ZAP Client-Side Input Validation Bypassing

Testcase 1:
Stored XSS Attack

Test Case	ASVS	Unique ID	CWE
1	5.3.3	5.3.3 - 1	79

ASVS 5.3.3:

Verify that context-aware, preferably automated - or at worst, manual – output escaping protects against reflected, stored, and DOM based XSS.

CWE-79: Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')

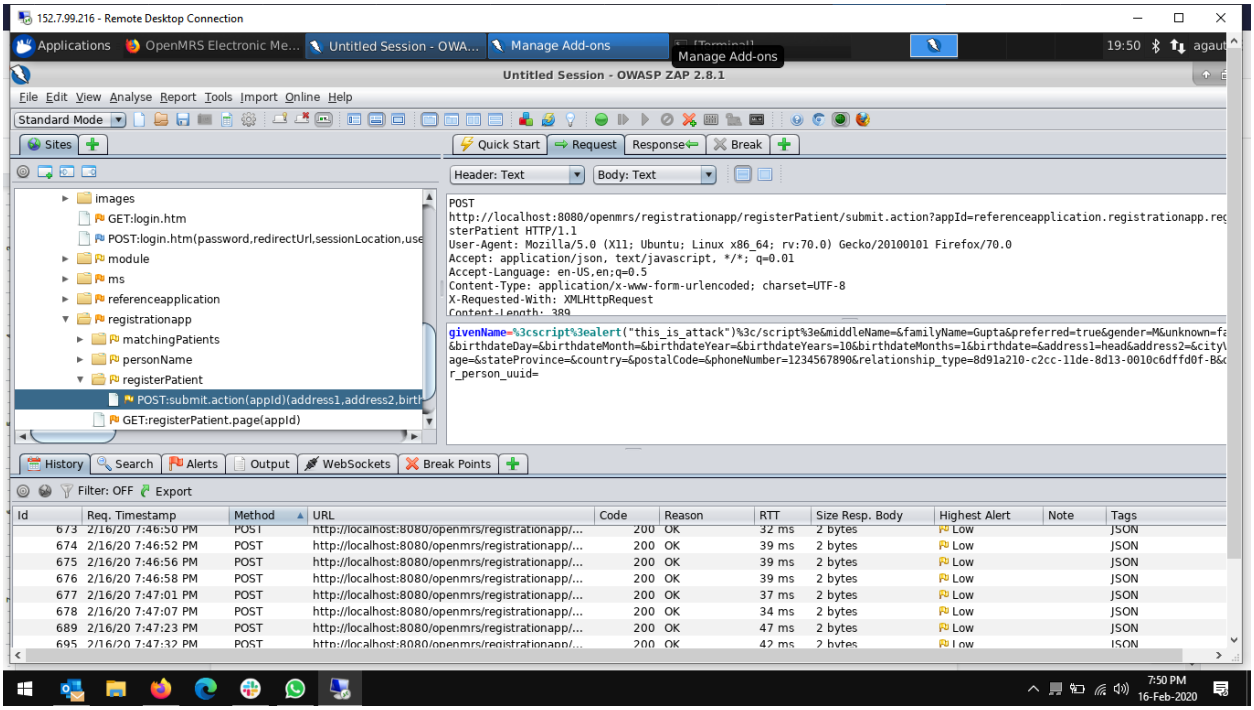
The software does not neutralize or incorrectly neutralizes user-controllable input before it is placed in output that is used as a web page that is served to other users.

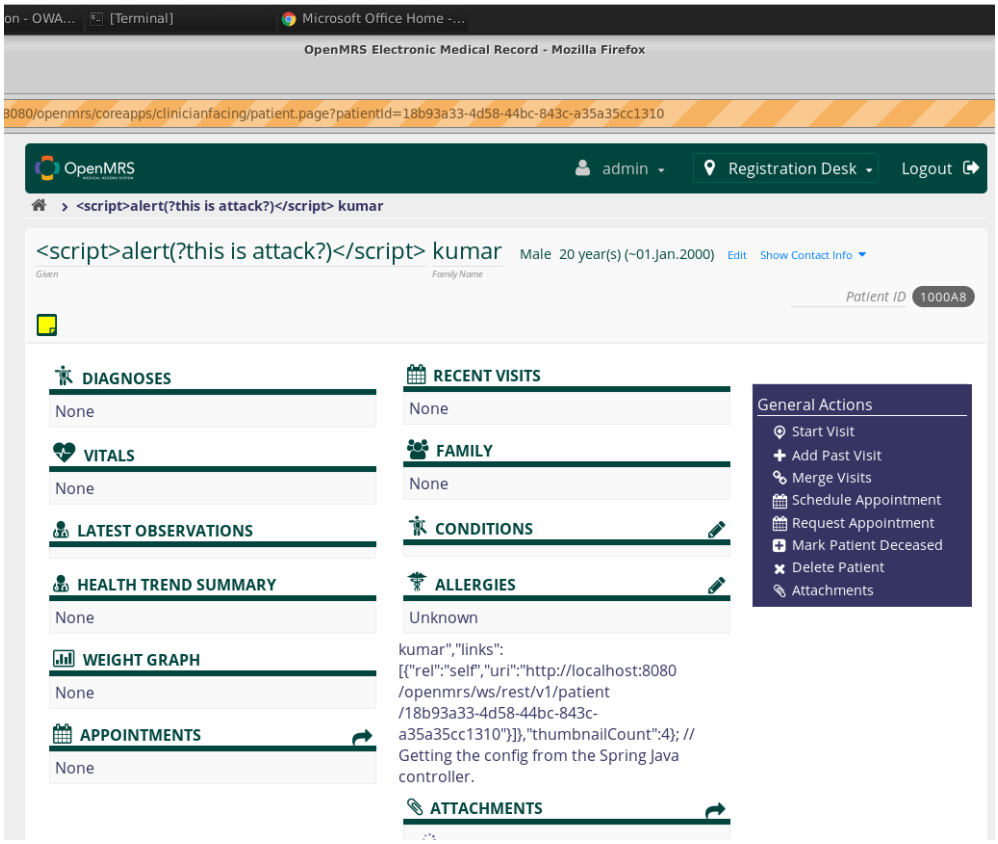
Repeatable Steps:

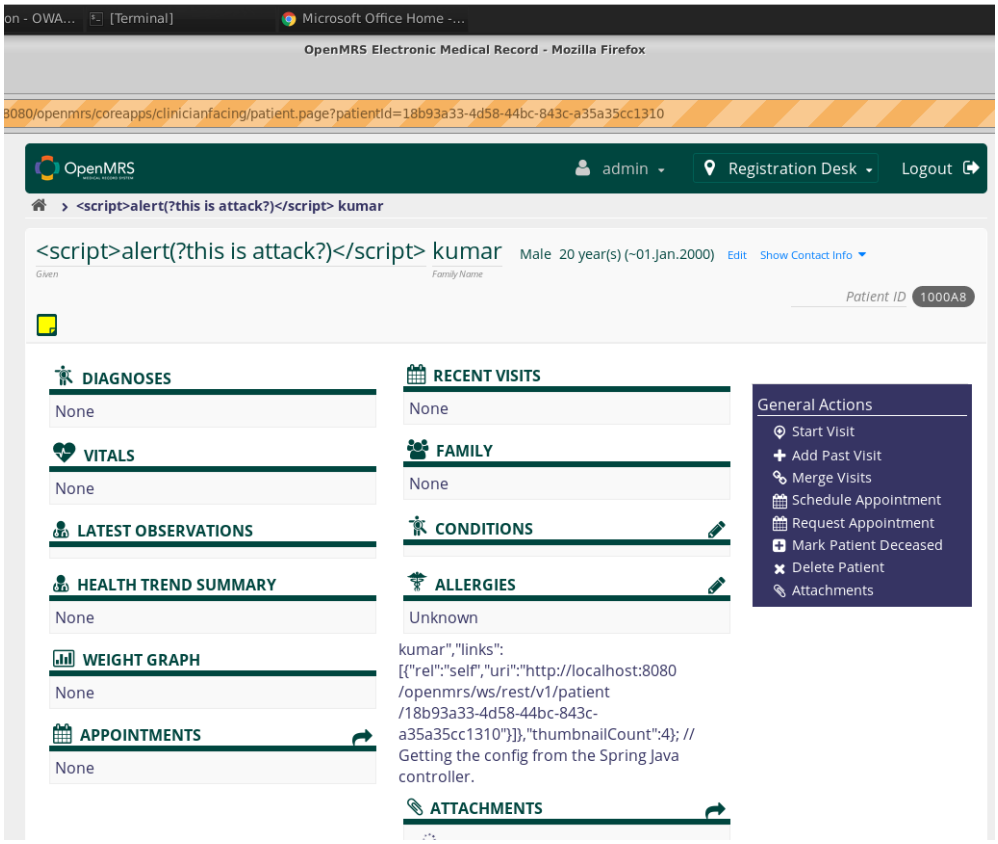
1. Navigate to OpenMRS login page – http://localhost:8080/openmrs/login.htm and open ZAP to capture web requests. You should see request to http://localhost:8080 under sites tab in ZAP.
2. Login to the application - Username: admin and Password: Admin123
3. Choose registration desk as location. Click on login.
4. Once logged in, click on 'Register a patient'.
5. Enter all the details:
   Given name: Raj
   Family name: Soni
   Gender: Male
   BirthDate: Day - 01 Month - Jan Year -1995
   Address: 123, abc st
   City:Raleigh
   State:NC
   Country:USA
   Postal Code:12345
   Phone Number: 123456789
   Select Relationship type: Sibling
   Person name: Mukesh
6. Click on confirm.
7. Go to ZAP, under Sites, navigate to http:localhost:8080 > openmrs >registrationapp > registerPatient module, you should find a POST request - POST:submit.action(appId)(address1...unknown)
8. Right click on it and add a breakpoint here.(Right click > break)

9. Go to browser, navigate to the register patient page - https://localhost:8080/openmrs/registrationapp/registerPatient.page?appId=referenceapplication.registrationapp.registerPatient
10. Follow step 5 but with a different given name – Roma2 and family name - Kumar
11. The site should stop at the set breakpoint when you click on confirm button.
12. Go to ZAP and update the given name in the POST body as - %3cscript%3ealert(”this is attack”)%3c/script%3e
13. Click on the play button twice in ZAP(jump to next breakpoint). Now we can see the response of that request on ZAP.
14. Navigate back to the OpenMRS application. You should be able to see that the given name is updated.

```
givenName=%3cscript%3ealert("this is attack")%3c/script%3e&middleName=&familyName=kumar&preferred=true&gender=M&unknown=false&birthdateDay=&birthdateMonth=&birthdateYear=&birthda
=20&birthdateMonths=&birthdate=&address1=123+st&address2=&cityVillage=raleigh&stateProvince=&country=&postalCode=123456&phoneNumber=1234567890&relationship_type=8d91a01c-c2cc-11d
0010c6dffd0f-A&other_person_uuid=
```




on - OWA...
[Terminal]
Microsoft Office Home -...
OpenMRS Electronic Medical Record - Mozilla Firefox
8080/openmrs/coreapps/clinicianfacing/patient.page?patientId=18b93a33-4d58-44bc-843c-a35a35cc1310
OpenMRS
admin
Registration Desk
Logout
<script>alert(?this is attack?)</script> kumar
<script>alert(?this is attack?)</script> kumar
Male 20 year(s) (~01.Jan.2000)
Edit
Show Contact Info
Given
Family Name
Patient ID
1000A8
DIAGNOSES
None
VITALS
None
LATEST OBSERVATIONS
HEALTH TREND SUMMARY
None
WEIGHT GRAPH
None
APPOINTMENTS
None
RECENT VISITS
None
FAMILY
None
CONDITIONS
ALLERGIES
Unknown
kumar","links":
[{"rel":"self","uri":"http://localhost:8080
/openmrs/ws/rest/v1/patient
/18b93a33-4d58-44bc-843c-
a35a35cc1310"}]},"thumbnailCount":4}; //
Getting the config from the Spring Java
controller.
ATTACHMENTS
General Actions
Start Visit
Add Past Visit
Merge Visits
Schedule Appointment
Request Appointment
Mark Patient Deceased
Delete Patient
Attachments



Initial User Input: Given Name - Roma2

Malicious Input: Given Name - 3cscript%3ealert(”this is attack”)%3c/script%3e

Actual Result:
Though we passed a script tag as input, it doesn't display any alert.

Expected Result:
No alert should be displayed

Testcase result:
PASS

Testcase 2:
SQL Injection:

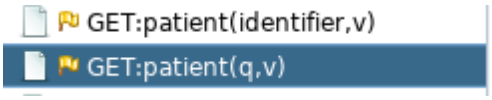
Test Case	ASVS	Unique ID	CWE
2	5.3.4	5.3.4-2	89

ASVS 5.3.4: Verify that data selection or database queries (e.g. SQL, HQL, ORM, NoSQL) use parameterized queries, ORMs, entity frameworks, or are otherwise protected from database injection attacks

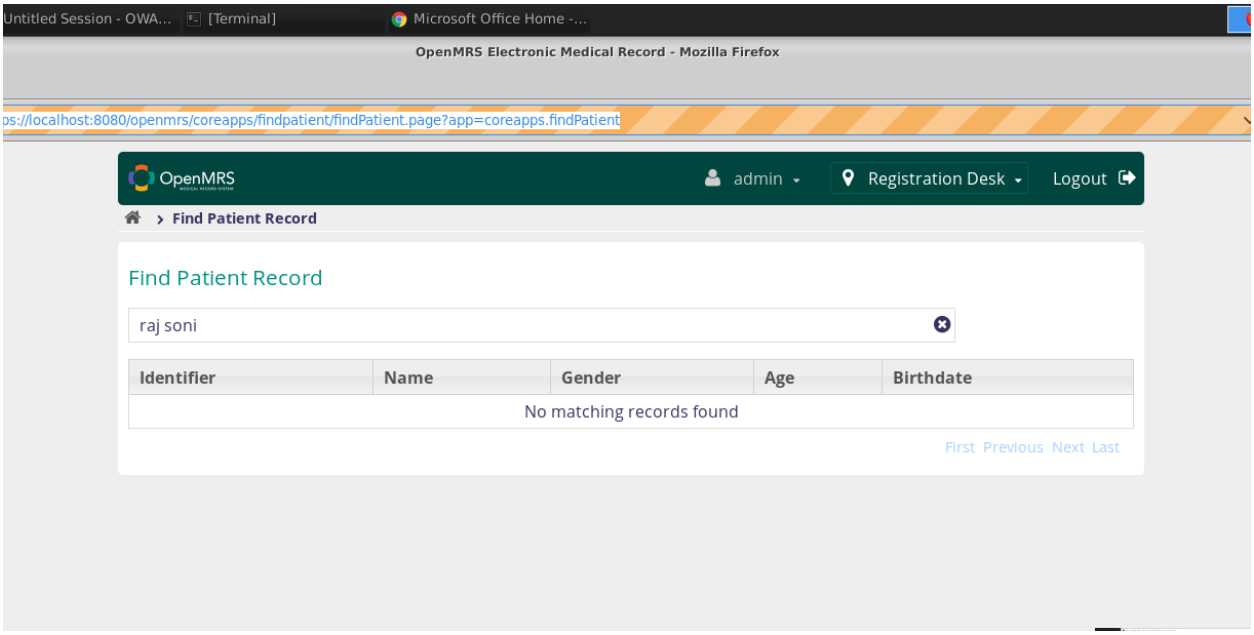
CWE 89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')
The software constructs all or part of an SQL command using externally influenced input from an upstream component, but it does not neutralize or

Repeatable Steps:

1. Navigate to OpenMRS login page – http://localhost:8080/openmrs/login.htm and open ZAP to capture web requests. You should see request to http://localhost:8080 under sites tab in ZAP.
2. Login to the application - Username: admin and Password: Admin123
3. Choose registration desk as location. Click on login.
4. Click on Find Patient record.
5. Enter 'Raj Soni' in the search field.
6. Go to ZAP, under Sites, navigate to http:localhost:8080 > openmrs > ws > rest > v1 module, you should find these GET requests – GET:patient(identifier,v) and GET:patient(q,v)
7. Right click on both of these requests and add a breakpoint here. (Right click > break)
8. Go to browser, navigate to the register patient page - https://localhost:8080/openmrs/coreapps/findpatient/findPatient.page?app=coreapps.findPatient
9. Enter 'Raj Soni'
10. The site should stop at the set breakpoint.
11. Go to ZAP, under Break Tab, modify the value of q query parameter to ' or 1=1--.
12. Click on the play button in ZAP (jump to next breakpoint). Now we can see the response of that request on ZAP.
13. Navigate back to the OpenMRS application. You should be able to see that no records are displayed.




```
GET http://localhost:8080/openmrs/ws/rest/v1/patient?q='or 1=1 --+soni&v=custom%3A(patientId%2Cuuid%2CpatientIdentifier%3A(uuid%2Cidentifier)%2Cperson%3A(gender%2Cage%2Cbirthdate%
2CbirthdateEstimated%2CpersonName)%2Cattributes%3A(value%2CattributeType%3A(name))) HTTP/1.1
Host: localhost:8080
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Disable-WWW-Authenticate: true
X-Requested-With: XMLHttpRequest
Connection: keep-alive
Referer: https://localhost:8080/openmrs/coreapps/findpatient/findPatient.page?app=coreapps.findPatient
Cookie: JSESSIONID=8lsz143bk3fzq0jw0mieuyjo; referenceapplication.lastSessionLocation=5; _REFERENCE_APPLICATION_LAST_USER_=92668751
If-None-Match: "03c36d6979f537fdebea95ebea2c99f63"
```



Initial User input: Find Patient - Raj Soni
Malicious input: Find Patient - ‘ or 1=1--

Expected Result:
Should not display any result.

Actual Result:
Does not display any result.

Result:
PASS

Testcase 3:
Buffer Overflow

Test Case	ASVS	Unique ID	CWE
3	5.4.1	5.4.1-3	120

ASVS 5.4.1:

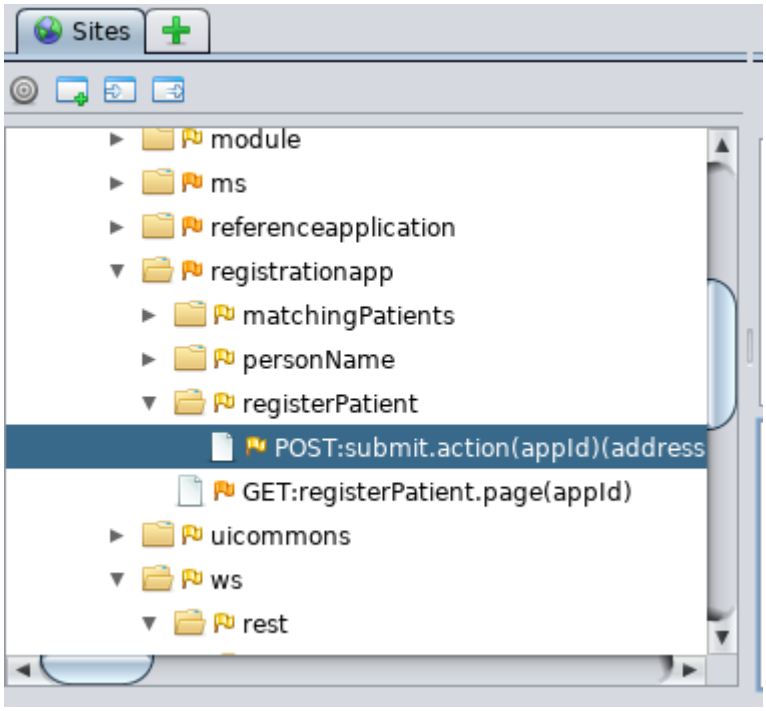
Verify that the application uses memory-safe string, safer memory copy and pointer arithmetic to detect or prevent stack, buffer, or heap overflows

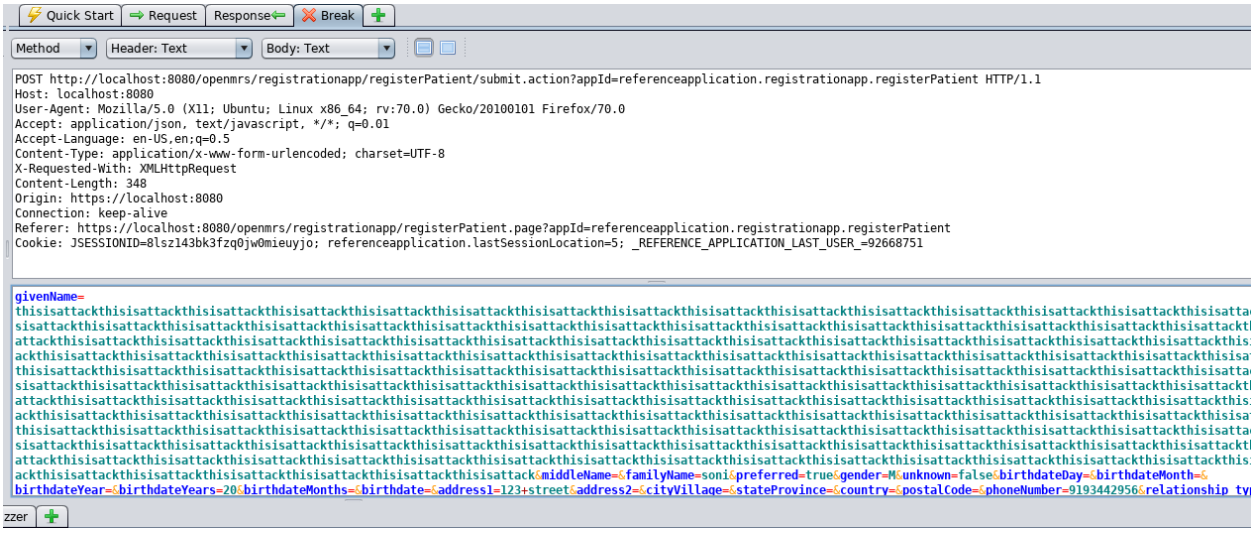
CWE-120: Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')

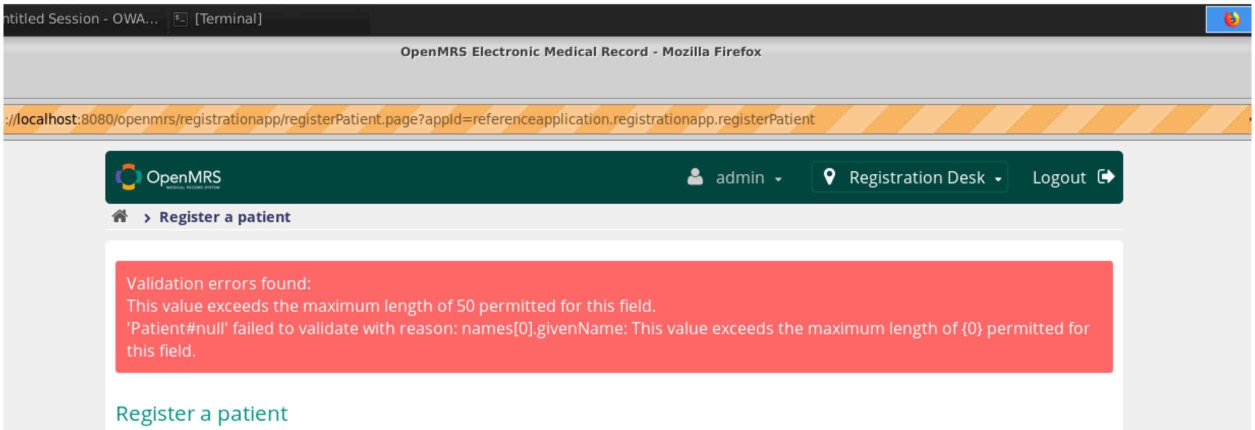
The program copies an input buffer to an output buffer without verifying that the size of the input buffer is less than the size of the output buffer, leading to a buffer overflow.

Repeatable Steps:

1. Navigate to OpenMRS login page – http://localhost:8080/openmrs/login.htm and open ZAP to capture web requests. You should see request to http://localhost:8080 under sites tab in ZAP.
2. Login to the application - Username: admin and Password: Admin123
3. Choose registration desk as location. Click on login.
4. Once logged in, click on 'Register a patient'.
5. Enter all the details:
   Given name: Raj
   Family name: Soni
   Gender: Male
   BirthDate: Day - 01 Month - Jan Year -1995
   Address: 123, abc st
   City:Raleigh
   State:NC
   Country:USA
   Postal Code:12345
   Phone Number: 123456789
   Select Relationship type: Sibling
   Person name: Mukesh
6. Click on confirm.
7. Go to ZAP, under Sites, navigate to http:localhost:8080 > openmrs >registrationapp > registerPatient module, you should find a POST request - POST:submit.action(appId)(address1...unknown)
8. Right click on it and add a breakpoint here.(Right click > break)
9. Go to browser, navigate to the register patient page - https://localhost:8080/openmrs/registrationapp/registerPatient.page?appId=referenceapplication.registrationapp.registerPatient
10. Follow step 5 but with a different given name – Ram and family name – Kumar
11. The site should stop at the set breakpoint when you click on confirm button.
12. Go to ZAP and update the given name in the POST body as - 'Thisisattackthisisattack
   thisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattack
   thisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattack'
13. Click on the play button twice in ZAP(jump to next breakpoint). Now we can see the response of that request on ZAP.
14. Navigate back to the OpenMRS application.


Sites
module
ms
referenceapplication
registrationapp
matchingPatients
personName
registerPatient
POST:submit.action(appId)(address
GET:registerPatient.page(appId)
uicommons
ws
rest


Quick Start
Request
Response
Break
Method
Header: Text
Body: Text
POST http://localhost:8080/openmrs/registrationapp/registerPatient/submit.action?appId=referenceapplication.registrationapp.registerPatient HTTP/1.1
Host: localhost:8080
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 348
Origin: https://localhost:8080
Connection: keep-alive
Referer: https://localhost:8080/openmrs/registrationapp/registerPatient.page?appId=referenceapplication.registrationapp.registerPatient
Cookie: JSESSIONID=8lsz143bk3fzq0jw0mieuyjo; referenceapplication.lastSessionLocation=5; _REFERENCE_APPLICATION_LAST_USER_=92668751
givenName=
birthdateYear=&birthdateYears=20&birthdateMonths=&birthdate=&address1=123+street&address2=&cityVillage=&stateProvince=&country=&postalCode=&phoneNumber=9193442956&relationship_ty



Initial User Input: Given Name - Ram
Malicious Input: Given Name **-**
Thisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisa
ttackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthi
sisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattac
k thisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattackthisisattack
thisisattackthisisattackthisisattackthisisattackthisisattackthisisattack

Expected Result:
The system should display an error message indicating that the length of the value exceeds the maximum length set.

Actual Result:
The system displays a validation error message.

Result:
PASS

Testcase 4:
Reflected XSS

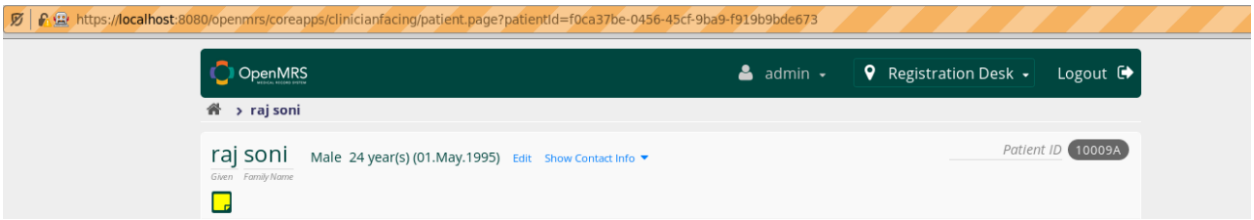
Test Case	ASVS	Unique ID	CWE
4	14.3.1	14.3.1-4	209

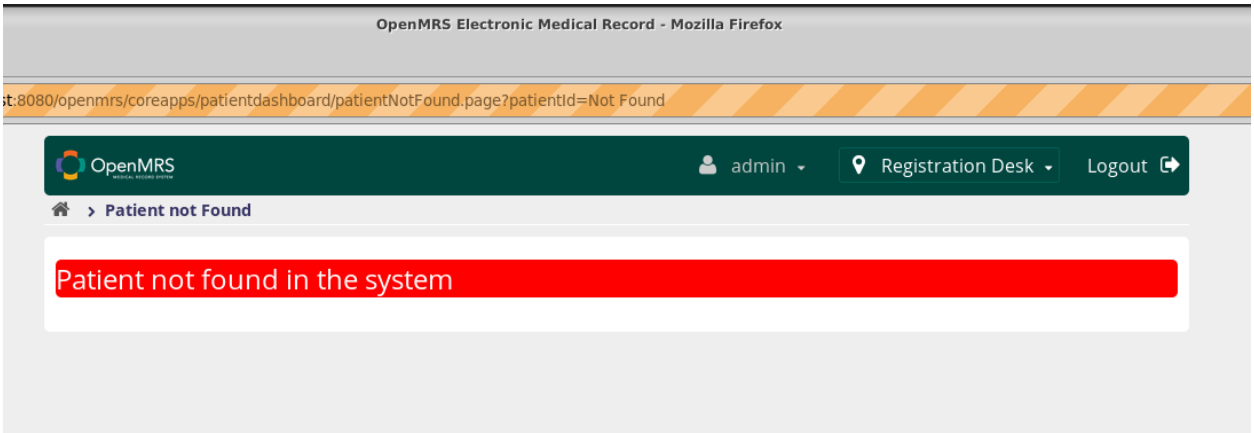
ASVS 14.3.1:
Verify that web or application server and framework error messages are configured to deliver user actionable, customized responses to eliminate any unintended security disclosures

CWE 209 : Information Exposure Through an Error Message
The software generates an error message that includes sensitive information about its environment, users, or associated data.

Repeatable Steps:

1. Navigate to OpenMRS login page – http://localhost:8080/openmrs/login.htm and open ZAP to capture web requests. You should see request to http://localhost:8080 under sites tab in ZAP.
2. Login to the application - Username: admin and Password: Admin123
3. Choose registration desk as location. Click on login.
4. Click on Find Patient record.
5. Enter 'Raj Soni' in the search field.
6. Open Raj Soni's record.
7. Go to ZAP, under Sites, navigate to http:localhost:8080 > openmrs > coreapps > clinicianfacing module, you should find these GET requests – GET:patient.page(patientID).
8. Right click on this request and add a breakpoint here. (Right click > break)
9. Go to browser, navigate to the register patient page - https://localhost:8080/openmrs/coreapps/findpatient/findPatient.page?app=coreapps.findPatient
10. Click on 'Raj Soni's' record
11. The site should stop at the set breakpoint.
12. Go to ZAP, under Break Tab, modify the value of patientID query parameter to %3cscript%3ealert("xss")%3c/script%3e
13. Click on the play button in ZAP (jump to next breakpoint). Now we can see the response of that request on ZAP.
14. Navigate back to the OpenMRS application.





Initial User input: Patient ID for Raj Soni in the URL- f0ca37be-0456-45cf-9ba9-f919b9bde673
Malicious user input: Patient ID for Raj Soni in the URL- %3cscript%3ealert(”xss”)%3c/script%3e

Expected Result:
Should not display patient's details

Actual Result:
Does not display the patient's details

Result:
PASS

Testcase 5:
XSS Attack

Test Case	ASVS	Unique ID	CWE
5	5.3.3	5.3.3-5	79

ASVS 5.3.3:

Verify that context-aware, preferably automated - or at worst, manual – output escaping protects against reflected, stored, and DOM based XSS.

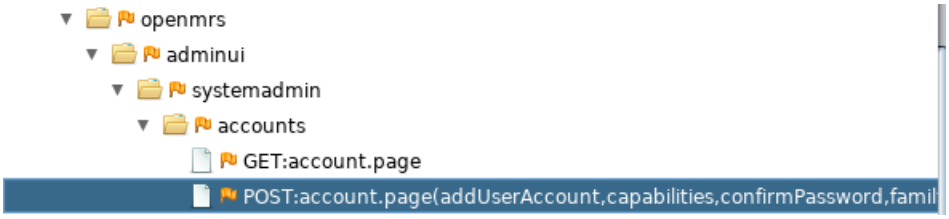
CWE-79: Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')

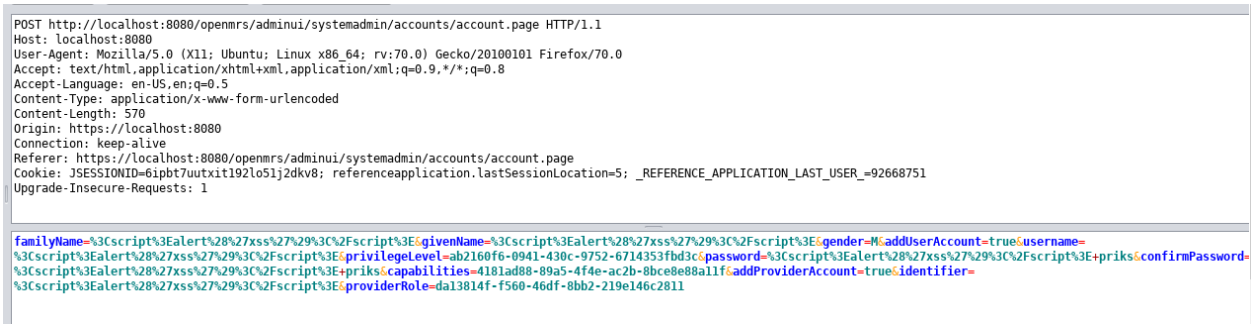
The software does not neutralize or incorrectly neutralizes user-controllable input before it is placed in output that is used as a web page that is served to other users.

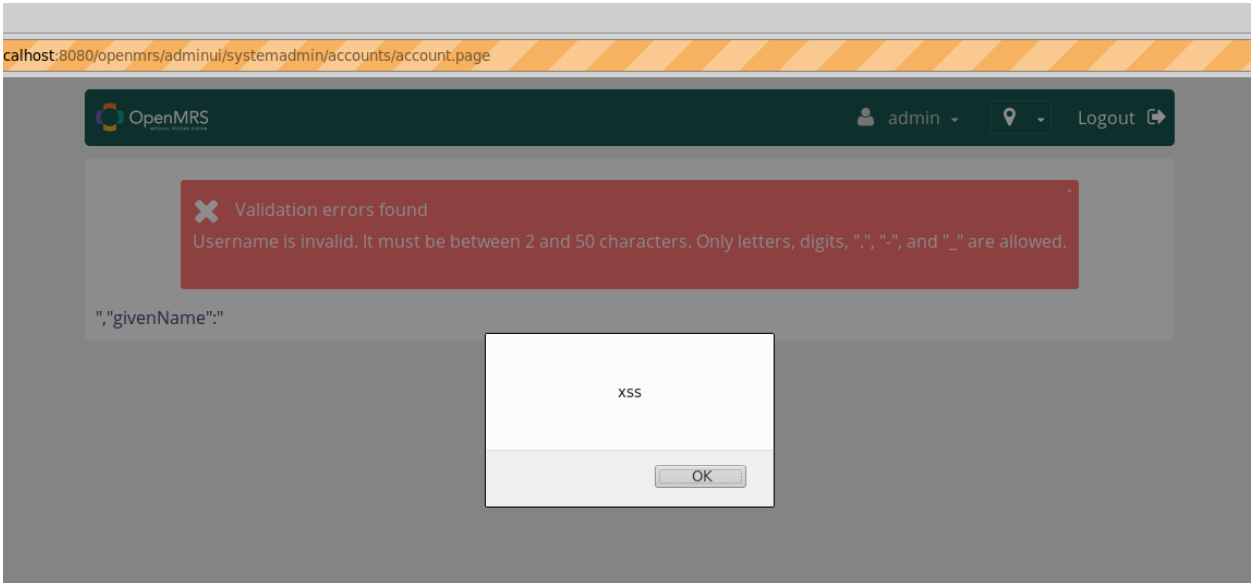
Repeatable Steps:

1. Navigate to OpenMRS login page – http://localhost:8080/openmrs/login.htm and open ZAP to capture web requests. You should see request to http://localhost:8080 under sites tab in ZAP.
2. Login to the application - Username: admin and Password: Admin123
3. Choose registration desk as location. Click on login.
4. Once logged in, click on 'System Administration'. Click on 'Manage accounts'.
5. Click on 'Add new account'.
6. Enter all the details:
   Family name: Roma
   Given name: roma
   Gender: Female
   Check on 'Add user account'
   Set username as Roma
   Privilege level – Full
   Password:12345678
   Confirm Password:12345678
   Uncheck force password change
   Check requests appointment
   Check Provider details
   Identifier – test
   Proivider Role: Doctor

7. Click on Save.
8. Go to ZAP, under Sites, navigate to http:localhost:8080 > openmrs >adminui> systemadmin > accounts module, you should find a POST request - POST:account.action(addUseraccount,….)
9. Right click on it and add a breakpoint here.(Right click > break)
   Go to browser, navigate to the add user account page - https://localhost:8080/openmrs/adminui/systemadmin/accounts/account.page
10. Follow step 5 & 6 but with a different given name – Ram and family name - Kumar
11. The site should stop at the set breakpoint when you click on Save button.
12. Go to ZAP and update the given name, family name, username, password, confirm password & identifier in the POST body as - %3cscript%3ealert('xss')%3c/script%3e
13. Click on the play button twice in ZAP(jump to next breakpoint). Now we can see the response of that request on ZAP.
14. Navigate back to the OpenMRS application.







Initial User Input – Given Name, Family Name, Username - Roma
Malicious user input - %3cscript%3ealert('xss')%3c/script%3e

Expected Result:
The script should not be executed
Actual Result:
The system does not discard <script> tag and executes it.

Testcase Result:
FAIL

Time Metrics for client-side bypassing:
--- total time taken for client-side bypassing: 3 hours
--- Total time to plan and run the 5 black box test cases : 3 hours (about 2 test case per hour approximately)
--- Total number of vulnerabilities found: 1 vulnerability

FUZZING - ZAP

Testcase 1:
Stored XSS Attack:

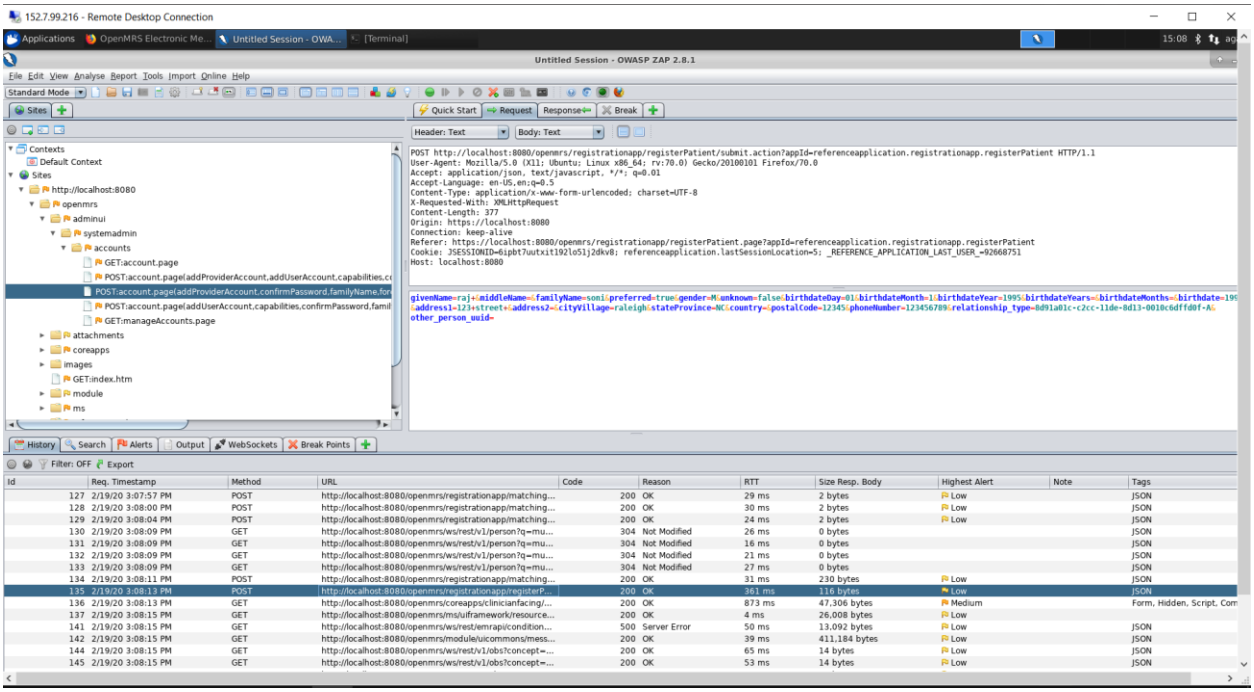
Test Case	ASVS	Unique ID	CWE
1	5.3.3	5.3.3-6	79

ASVS 5.3.3:
Verify that context-aware, preferably automated - or at worst, manual – output escaping protects against reflected, stored, and DOM based XSS.

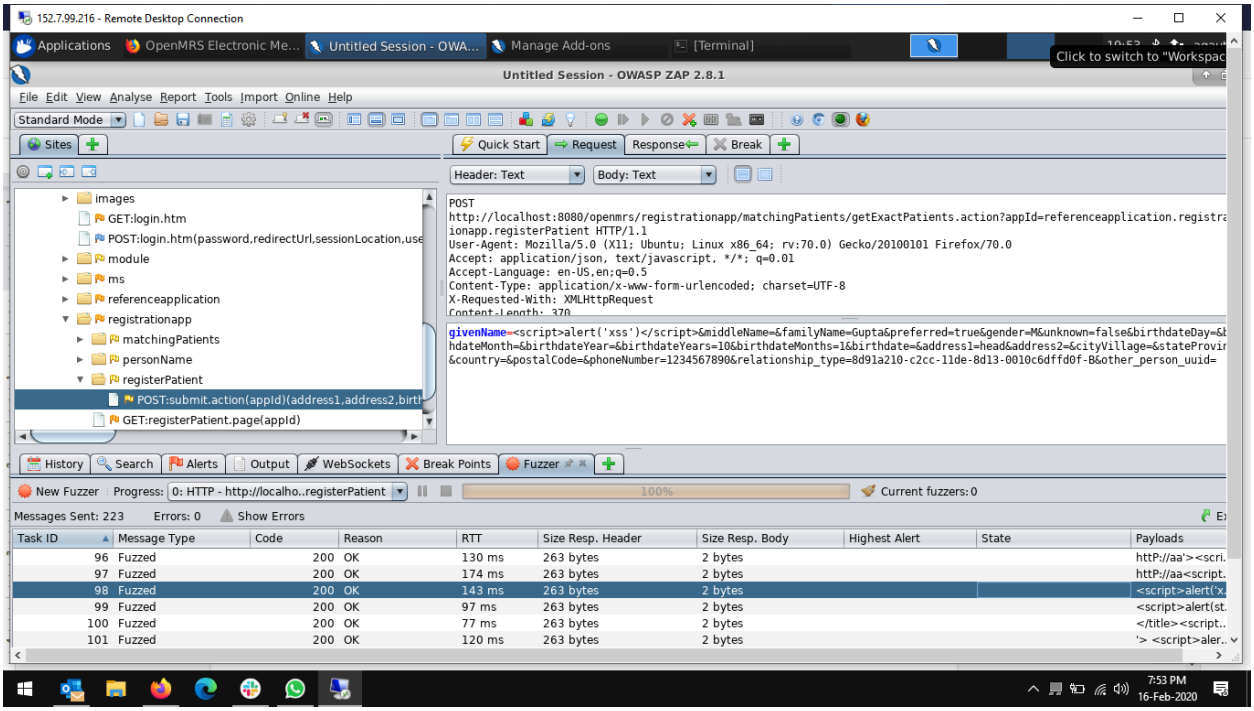
CWE-79: Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')
The software does not neutralize or incorrectly neutralizes user-controllable input before it is placed in output that is used as a web page that is served to other users.

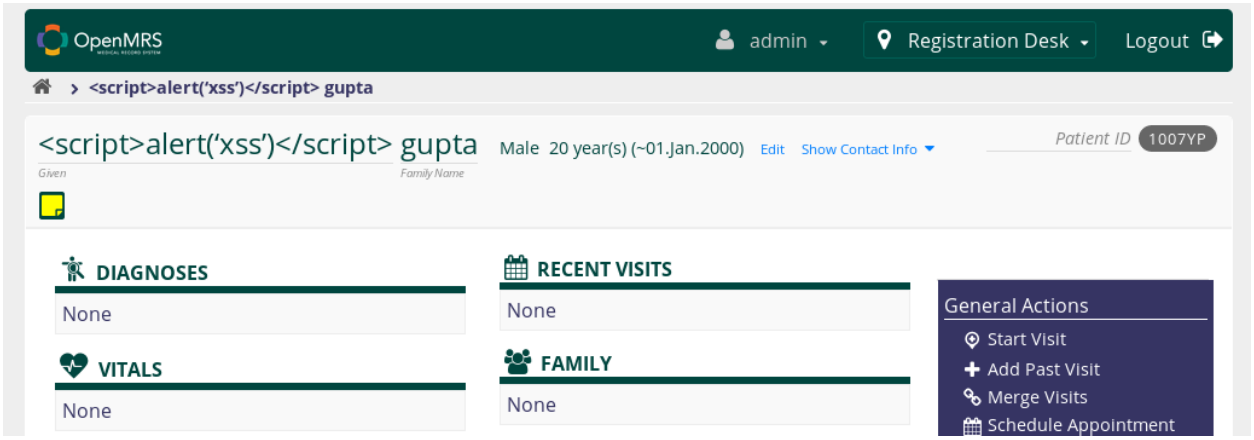
Repeatable Steps:

1. Navigate to OpenMRS login page – http://localhost:8080/openmrs/login.htm and open ZAP to capture web requests. You should see request to http://localhost:8080 under sites tab in ZAP.
2. Login to the application - Username: admin and Password: Admin123
3. Choose registration desk as location. Click on login.
4. Once logged in, click on 'Register a patient'.
5. Enter all the details:
   Given name: Raj
   Family name: Soni
   Gender: Male
   BirthDate: Day - 01 Month - Jan Year -1995
   Address: 123, abc st
   City:Raleigh
   State:NC
   Country:USA
   Postal Code:12345
   Phone Number: 123456789
   Select Relationship type: Sibling
   Person name: Mukesh
6. Click on confirm.
7. Go to ZAP, get the POST request.



8. Double click on the given name value which is Raj in this case. Right click > click on fuzz > click on payloads > click on Add > Set the type to File fuzzers > Under jbrofuzz > Select XSS > Click add > Click OK > Start fuzzer.
9. Choose one of the fuzzed result - <script>alert('xss')</script>
10. Use this value as given name while registering a patient in OpenMRS application.





Actual Result:
Adding this value in the UI does not give any alert.

Expected Result:
Script tag should not be executed.

Result:
PASS

The application does not execute the script tag when it is given as an input, thus showing that the input is validated. Code differentiates between data and executable statements.

Testcase 2:

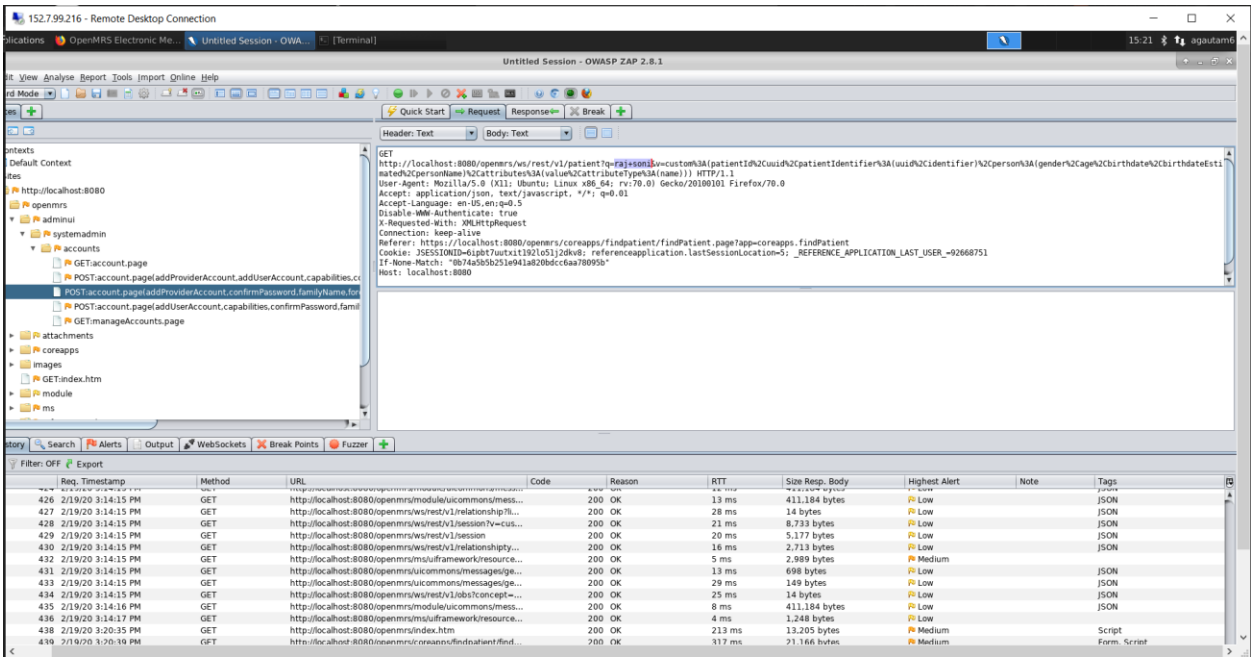
SQL Injection

Test Case	ASVS	Unique ID	CWE
2	5.3.4	5.3.4-7	89

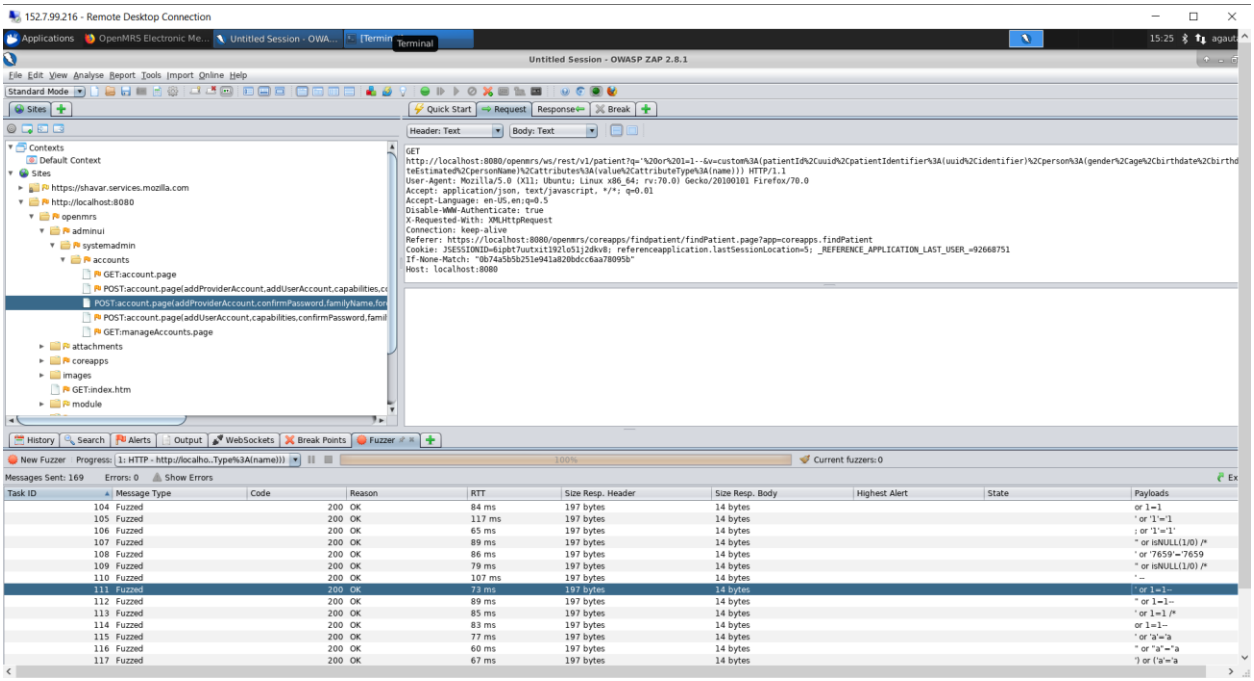
ASVS 5.3.4: Verify that data selection or database queries (e.g. SQL, HQL, ORM, NoSQL) use parameterized queries, ORMs, entity frameworks, or are otherwise protected from database injection attacks
CWE 89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')
The software constructs all or part of an SQL command using externally influenced input from an upstream component, but it does not neutralize or

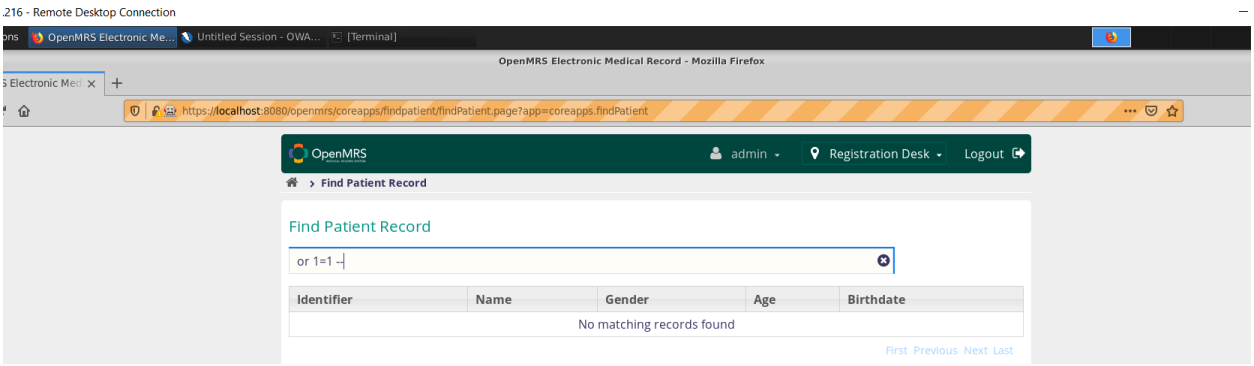
Repeatable Steps:

1. Navigate to OpenMRS login page – http://localhost:8080/openmrs/login.htm and open ZAP to capture web requests. You should see request to http://localhost:8080 under sites tab in ZAP.
2. Login to the application - Username: admin and Password: Admin123
3. Choose registration desk as location. Click on login.
4. Click on Find Patient record.
5. Enter 'Raj Soni' in the search field.
6. Go to ZAP, choose the request as shown below:



7. Double click on the value of q query parameter, right click > click on fuzz > click on payloads > click on Add > Set the type to File fuzzers > Under jbrofuzz > Select SQL Injection> Click add > Click OK > Start fuzzer.
11. Choose one of the fuzzed result – “or 1=1--”
12. Use this value as given name while registering a patient.
    Use this value while finding a patient record in OpenMRS application.





Expected Result:
The system should not display all results

Actual Result:
Does not display any result, indicates that no records are found

Testcase Result:
PASS

Since, the attack was not successful it shows that the system uses a mechanism such as prepared statements to tackle SQL injection vulnerabilities. Prepared statements are a standard feature in development libraries now.

Testcase 3:
Buffer Overflow

Test Case	ASVS	Unique ID	CWE
3	5.4.1	5.4.1-8	120

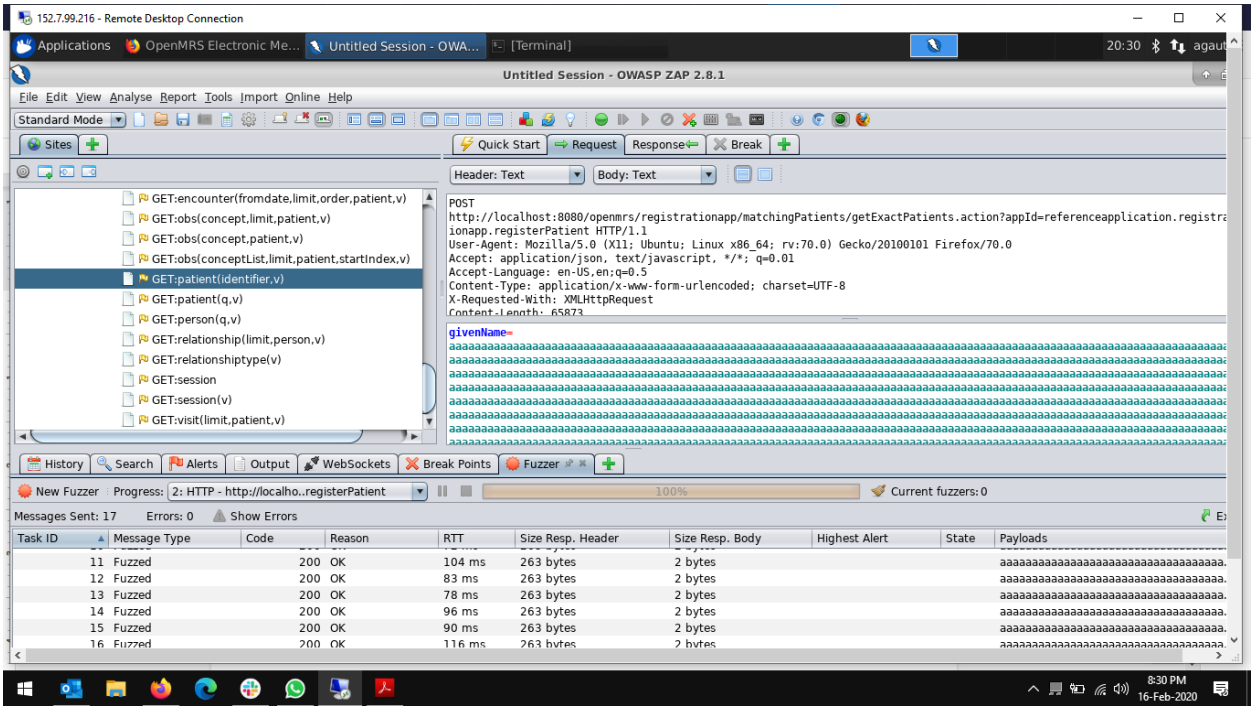
ASVS 5.4.1:
Verify that the application uses memory-safe string, safer memory copy and pointer arithmetic to detect or prevent stack, buffer, or heap overflows
Repeatable Steps:

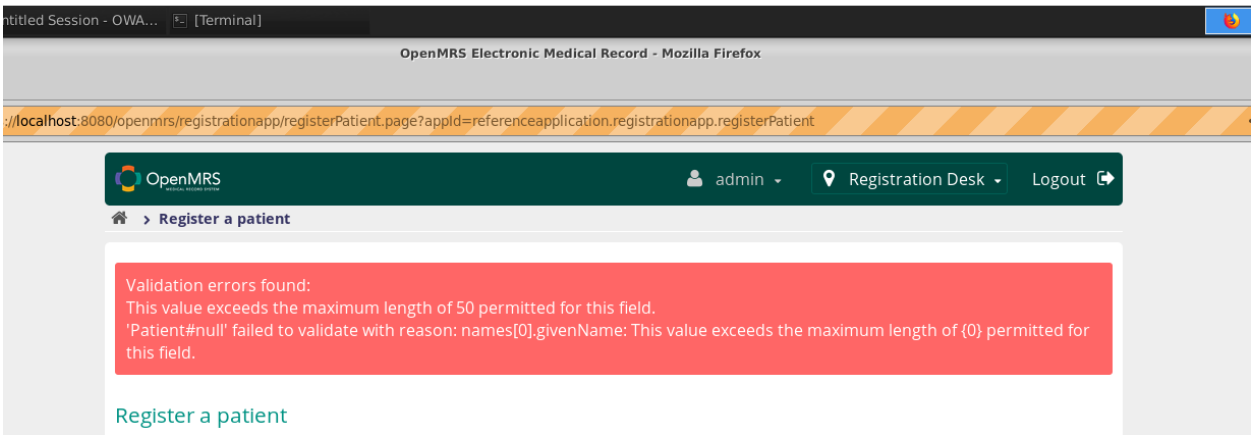
CWE-120: Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')
The program copies an input buffer to an output buffer without verifying that the size of the input buffer is less than the size of the output buffer, leading to a buffer overflow.

Repeatable Steps:

1. Navigate to OpenMRS login page – http://localhost:8080/openmrs/login.htm and open ZAP to capture web requests. You should see request to http://localhost:8080 under sites tab in ZAP.
2. Login to the application - Username: admin and Password: Admin123
3. Choose registration desk as location. Click on login.
4. Once logged in, click on 'Register a patient'.
5. Enter all the details:
   Given name: Raj
   Family name: Soni
   Gender: Male
   BirthDate: Day - 01 Month - Jan Year -1995
   Address: 123, abc st
   City:Raleigh
   State:NC
   Country:USA
   Postal Code:12345
   Phone Number: 123456789
   Select Relationship type: Sibling
   Person name: Mukesh

6. Click on confirm.
7. Go to ZAP, get the POST request.
8. Double click on the given name value which is Raj in this case. Right click > click on fuzz > click on payloads > click on Add > Set the type to File zzers > Under jbrofuzz > Select Buffer Overflow > Click add > Click OK > Start fuzzer.
9. Choose one of the fuzzed result – aaaaaaaaaaaaaaaaaaaaa.....
   Use this value as given name while registering a patient in OpenMRS application.





Expected Result:
Should display an error message. Request should not be successful

Actual Result:
Application displays an error message

Result:
PASS

This shows that application has implemented techniques to mitigate buffer overflow attack and also there is a limit on the maximum nuber of input characters that can be entered.

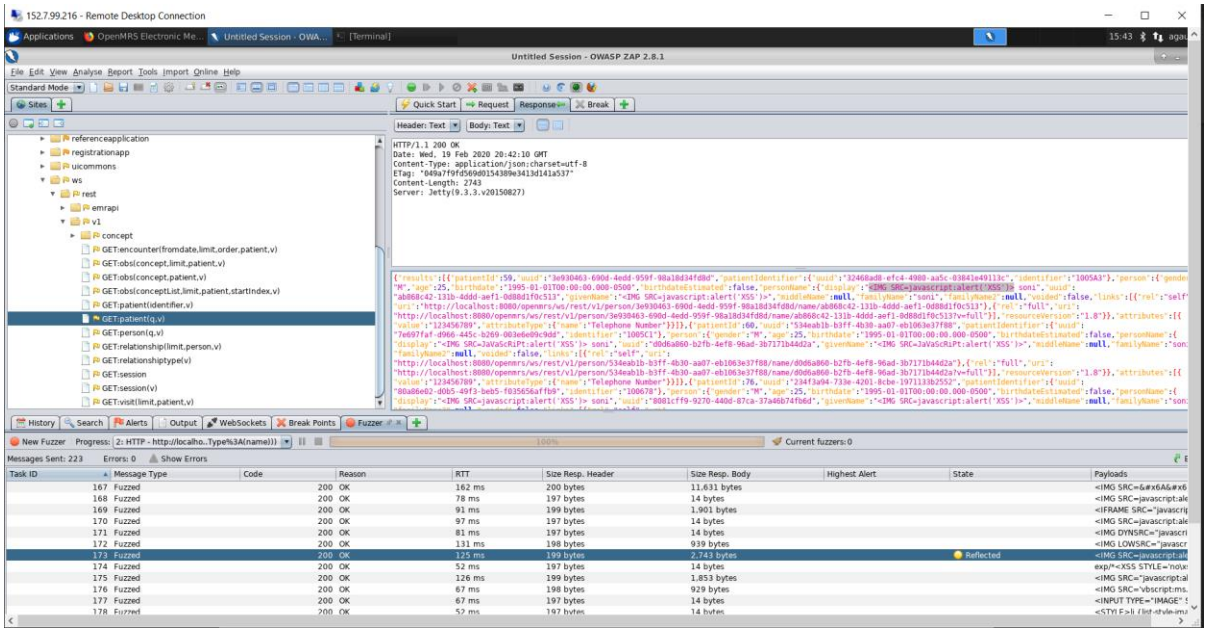
Testcase 4:
Reflected XSS

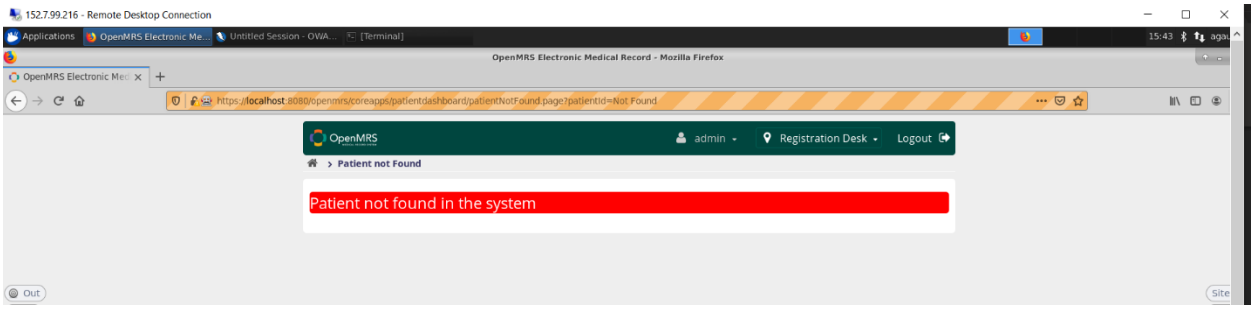
Test Case	ASVS	Unique ID	CWE
4	14.3.1	14.3.1-9	209

ASVS 14.3.1:
Verify that web or application server and framework error messages are configured to deliver user actionable, customized responses to eliminate any unintended security disclosures
CWE 209 : Information Exposure Through an Error Message
The software generates an error message that includes sensitive information about its environment, users, or associated data.

Repeatable steps:

1. Navigate to OpenMRS login page – http://localhost:8080/openmrs/login.htm and open ZAP to capture web requests. You should see request to http://localhost:8080 under sites tab in ZAP.
2. Login to the application - Username: admin and Password: Admin123
3. Choose registration desk as location. Click on login.
4. Click on Find Patient record.
5. Enter 'Raj Soni' in the search field.
6. Open Raj's record. URL would be - https://localhost:8080/openmrs/coreapps/clinicianfacing/patient.page?patientId=17eeff14-bc5f-4d61-8b12-6379dd50b3f3
7. Go to ZAP, choose the request - https://localhost:8080/openmrs/coreapps/clinicianfacing/patient.page?patientId=17eeff14-bc5f-4d61-8b12-6379dd50b3f3
8. Double click on - 17eeff14-bc5f-4d61-8b12-6379dd50b3f3
9. Right click > click on fuzz > click on payloads > click on Add > Set the type to File zzers > Under jbrofuzz > Select XSS > Click add > Click OK > Start fuzzer.
10. Choose one of the fuzzed result.
11. Use this value as name while finding a patient in OpenMRS application.





Expected Result:
Should not display any result

Actual Result:
Does not display any patients record.

Testcase result:
PASS

Since the attack was not successful, it shows the application validates the input/escapes input characters and does not execute the script tags in the URL.

Testcase 5:

Test Case	ASVS	Unique ID	CWE

5	5.3.3	5.3.3-10	79

ASVS 5.3.3:

Verify that context-aware, preferably automated - or at worst, manual – output escaping protects against reflected, stored, and DOM based XSS.

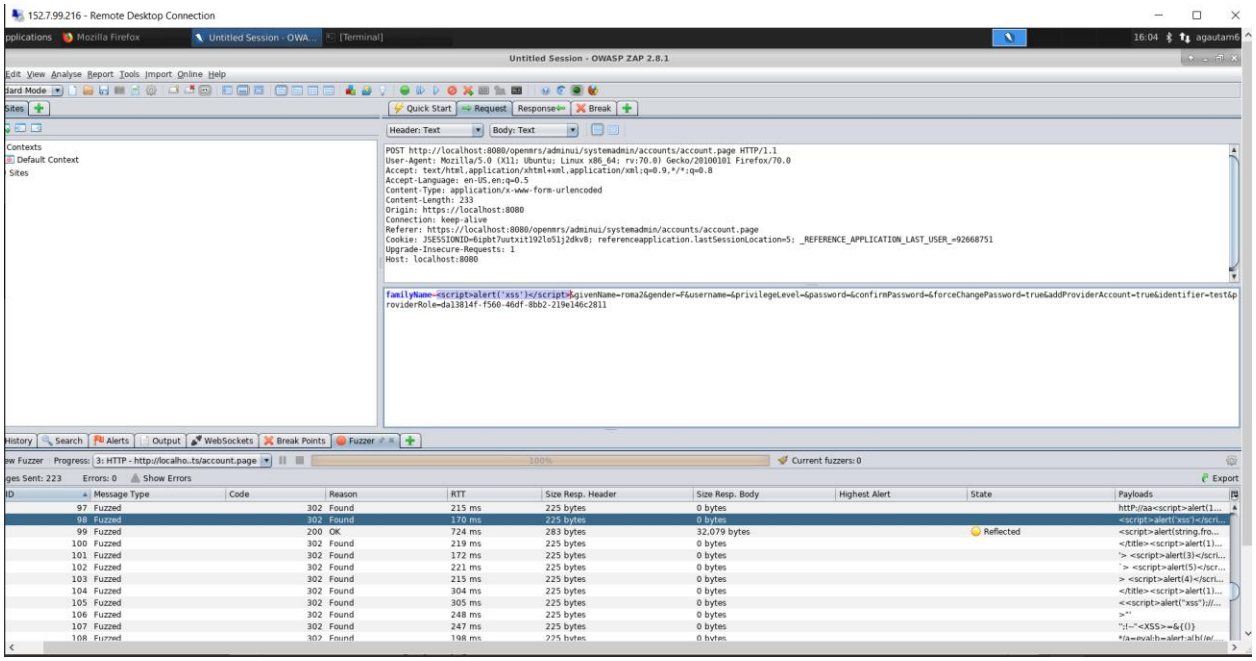
CWE-79: Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')

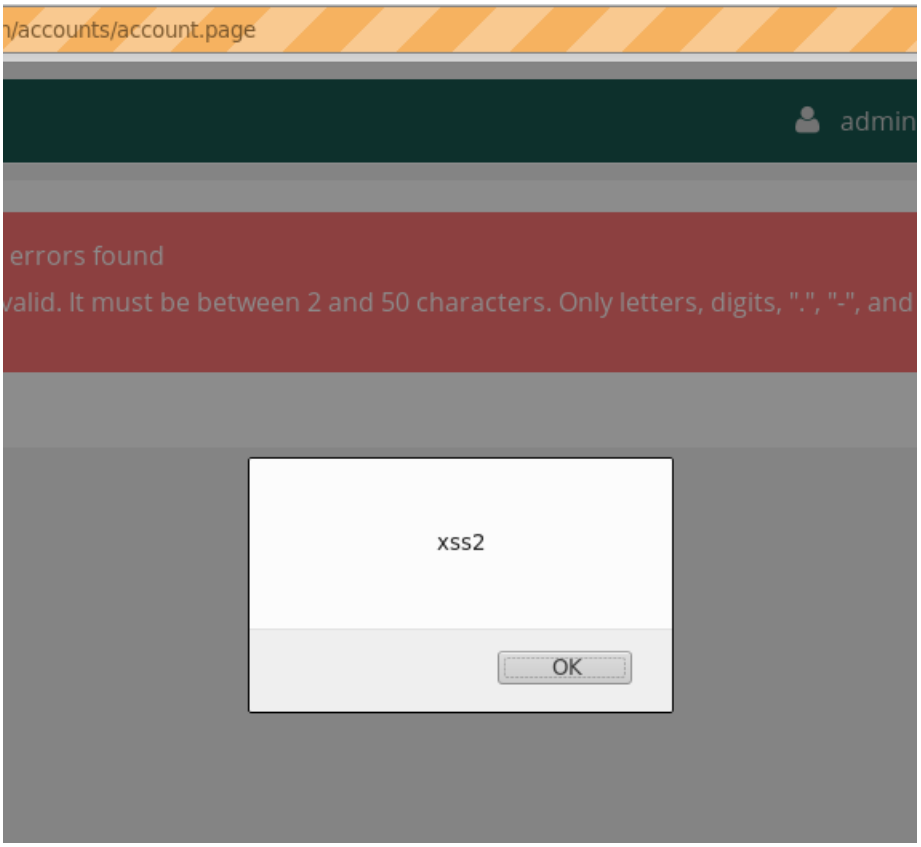
The software does not neutralize or incorrectly neutralizes user-controllable input before it is placed in output that is used as a web page that is served to other users.

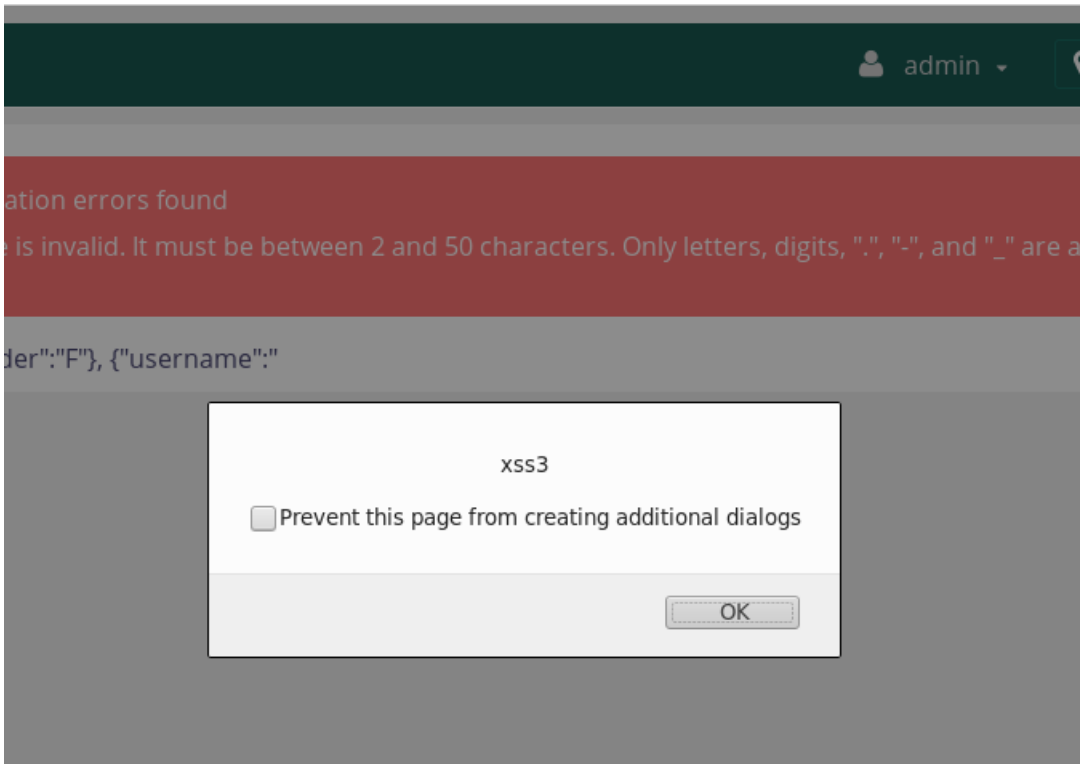
Repeatable Steps:

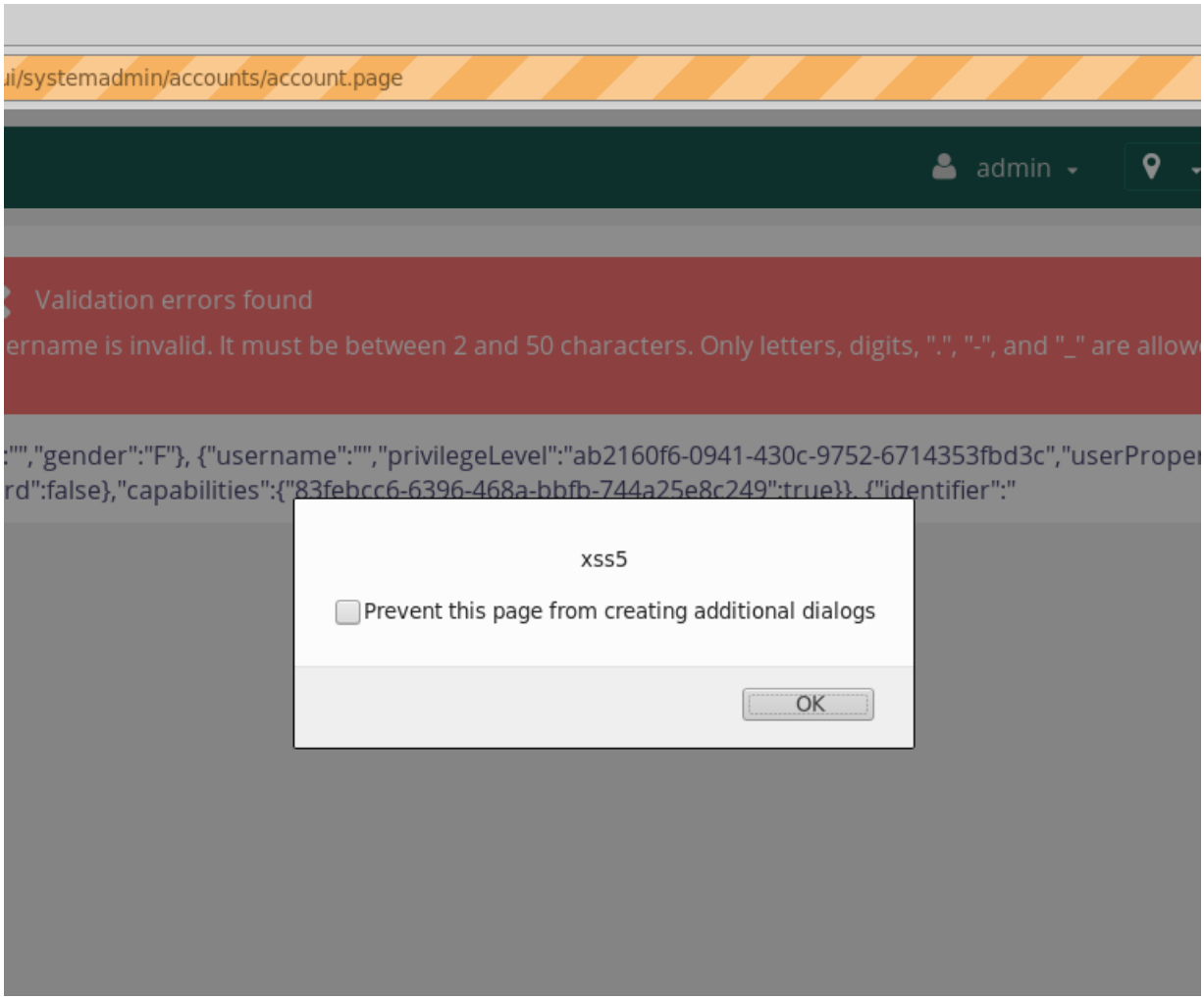
1. Navigate to OpenMRS login page – http://localhost:8080/openmrs/login.htm and open ZAP to capture web requests. You should see request to http://localhost:8080 under sites tab in ZAP.Login to the application - Username: admin and Password: Admin123
2. Choose registration desk as location. Click on login.
3. Once logged in, click on ‘System Administration’. Click on ‘Manage accounts’.
4. Click on ‘Add new account’.
5. Enter all the details:
   Family name: Roma
   Given name: roma
   Gender: Female
   Check on ‘Add user account’
   Set username as Roma
   Privilege level – Full
   Password:12345678
   Confirm Password:12345678
   Uncheck force password change
   Check requests appointment
   Check Provider details
   Identifier – test
   Proivider Role: Doctor

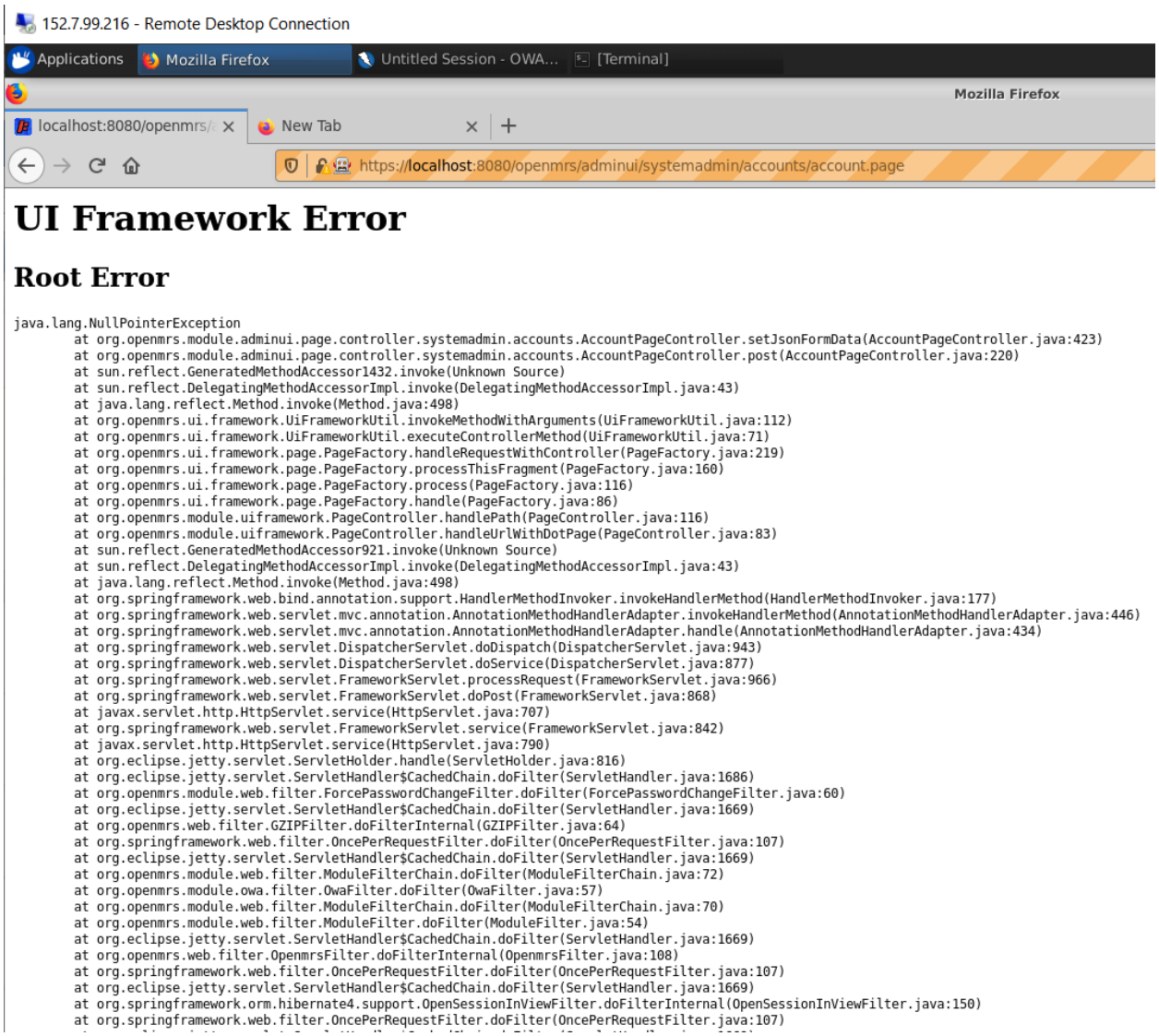
6. Click on Save.
7. Go to ZAP, get the POST request.
8. Double click on the given name value which is Raj in this case. Right click > click on fuzz > click on payloads > click on Add > Set the type to File fuzzers > Under jbrofuzz > Select XSS > Click add > Click OK > Start fuzzer.
9. Choose one of the fuzzed result - <script>alert(‘xss’)</script>
10. Use this value as given name, family name, username and identifier while adding a new user in OpenMRS application.


152.7.99.216 - Remote Desktop Connection
Untitled Session - OWASP ZAP 2.8.1
POST http://localhost:8080/openmrs/adminui/systemadmin/accounts/account.page HTTP/1.1
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Content-Type: application/x-www-form-urlencoded
Content-Length: 233
Origin: https://localhost:8080
Connection: keep-alive
Referer: https://localhost:8080/openmrs/adminui/systemadmin/accounts/account.page
Cookie: JSESSIONID=6ipbt7autxit192lo51j2dkv8; referenceapplication.lastSessionLocation=5; _REFERENCE_APPLICATION_LAST_USER_=92668751
Upgrade-Insecure-Requests: 1
Host: localhost:8080
familyName=<script>alert('xss')</script>&givenName=roma2&gender=F&username=&privilegeLevel=&password=&confirmPassword=&forceChangePassword=true&addProviderAccount=true&identifier=test&providerRole=da13814f-f560-46df-8bb2-219e146c2811


/accounts/account.page
admin
errors found
valid. It must be between 2 and 50 characters. Only letters, digits, ".", "-", and
xss2
OK


admin
ation errors found
is invalid. It must be between 2 and 50 characters. Only letters, digits, ".", "-", and "_" are a
der":"F"}, {"username":"
xss3
Prevent this page from creating additional dialogs
OK


ui/systemadmin/accounts/account.page
admin
Validation errors found
ername is invalid. It must be between 2 and 50 characters. Only letters, digits, ".", "-", and "_" are allow
:"","gender":"F"}, {"username":"","privilegeLevel":"ab2160f6-0941-430c-9752-6714353fbd3c","userPrope
rd":false},"capabilities":{"83febcc6-6396-468a-bbfb-744a25e8c249":true}}, {"identifier":"
xss5
Prevent this page from creating additional dialogs
OK



Expected Result:
Should not execute the script and should not give stack trace.

Actual Result:
It executes the script. Also, clicking on cancel gives the stack trace.

Testcase result:
FAIL

The attack was successful, showing that application is vulnerable to XSS attack.

Testcase 6:
Injection

ASVS 5.3.4: Verify that data selection or database queries (e.g. SQL, HQL, ORM, NoSQL) use parameterized queries, ORMs, entity frameworks, or are otherwise protected from database injection attacks

CWE 89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')
The software constructs all or part of an SQL command using externally influenced input from an upstream component, but it does not neutralize or

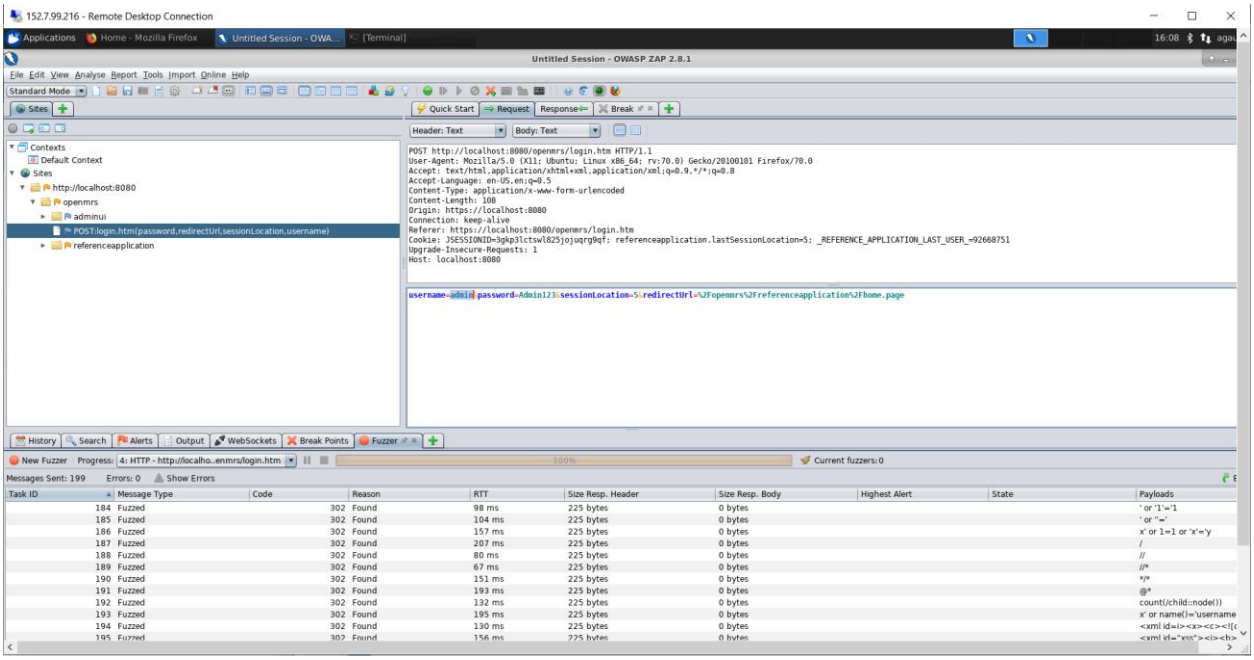
Test Case	ASVS	Unique ID	CWE
6	5.3.4	5.3.4-11	89

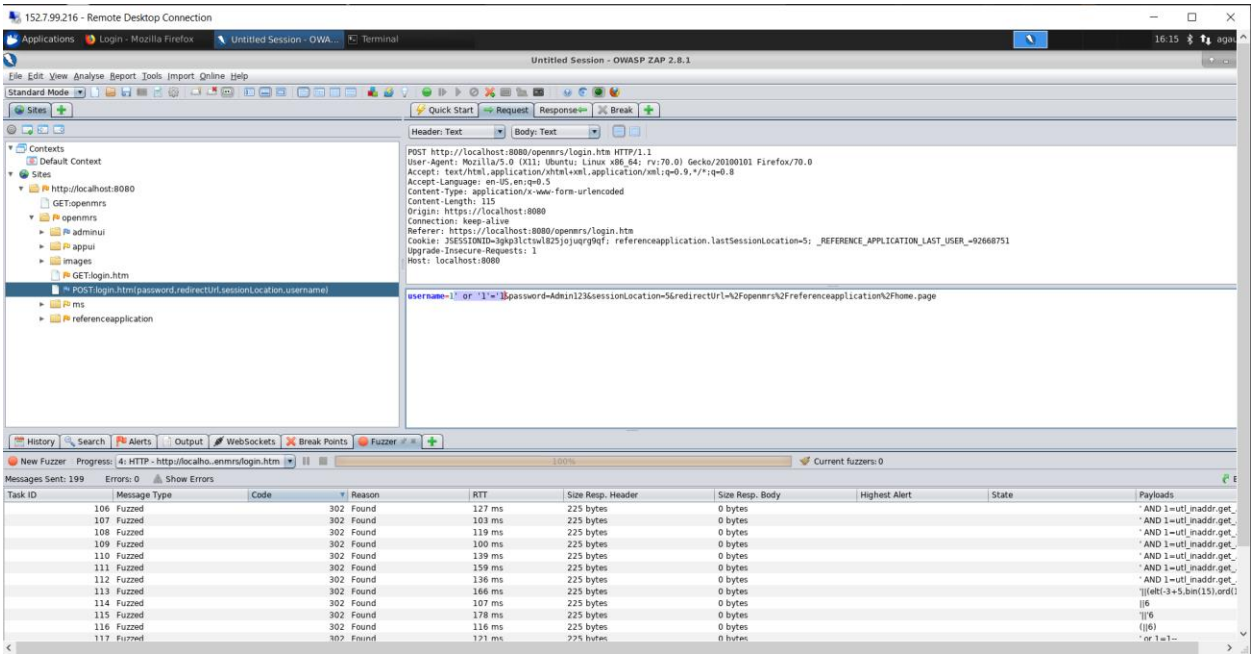
ASVS 5.3.4: Verify that data selection or database queries (e.g. SQL, HQL, ORM, NoSQL) use parameterized queries, ORMs, entity frameworks, or are otherwise protected from database injection attacks

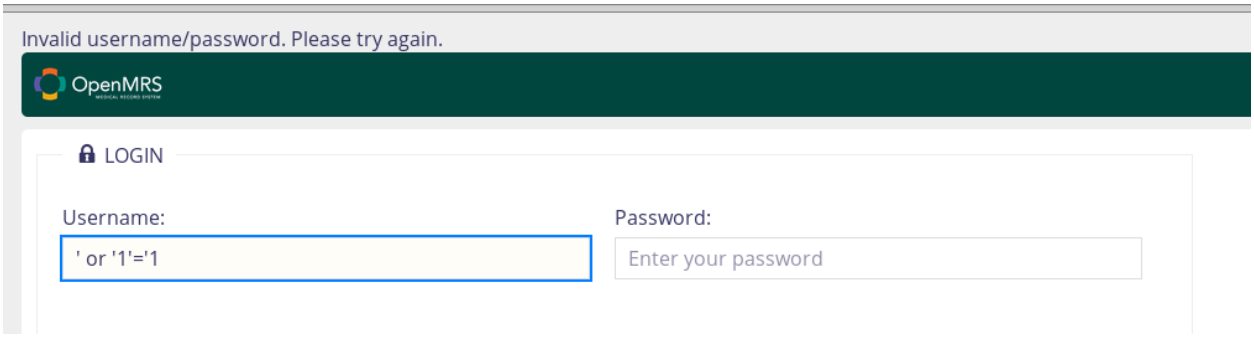
CWE 89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')
The software constructs all or part of an SQL command using externally influenced input from an upstream component, but it does not neutralize or

Repeatable Steps:

1. Navigate to OpenMRS login page – http://localhost:8080/openmrs/login.htm and open ZAP to capture web requests. You should see request to http://localhost:8080 under sites tab in ZAP.
2. Login to the application - Username: admin and Password: Admin123
3. Choose registration desk as location. Click on login.
4. Go to ZAP, get the POST request.
5. Double click on the given name value which is Raj in this case. Right click > click on fuzz > click on payloads > click on Add > Set the type to File fuzzers > Under jbrofuzz > Select Injection> Click add > Click OK > Start fuzzer.
6. Choose one of the fuzzed result -
7. Login to OpenMRS application using this value


152.7.99.216 - Remote Desktop Connection
Untitled Session - OWASP ZAP 2.8.1
POST http://localhost:8080/openmrs/login.htm HTTP/1.1
username=admin&password=Admin123&sessionLocation=5&redirectUrl=%2Fopenmrs%2Freferenceapplication%2Fhome.page


152.7.99.216 - Remote Desktop Connection
Untitled Session - OWASP ZAP 2.8.1
username=1' or '1'='1&password=Admin123&sessionLocation=5&redirectUrl=%2Fopenmrs%2Freferenceapplication%2Fhome.page


Invalid username/password. Please try again.
OpenMRS
LOGIN
Username:
' or '1'='1
Password:
Enter your password

Expected Result:
Should not be able to login

Actual Result:
Does not let you login, displays an error message.

Testcase Result:
PASS

This shows that login page is not vulnerable to SQL injection attack, because the input is validated.

Time Metrics for Fuzzing:
--- total time taken for client-side bypassing: 3 hours
--- Total time to plan and run the 6 black box test cases: 3 hours (about 2 test case per hour approximately)
--- Total number of vulnerabilities found: 1 vulnerability

2. Defensics

Test Case1:

ASVS : 13.1.3:Verify API URLs do not expose sensitive information, such as the API key, session tokens etc.

CWE-598: Information Exposure Through Query Strings in GET Request

Steps:

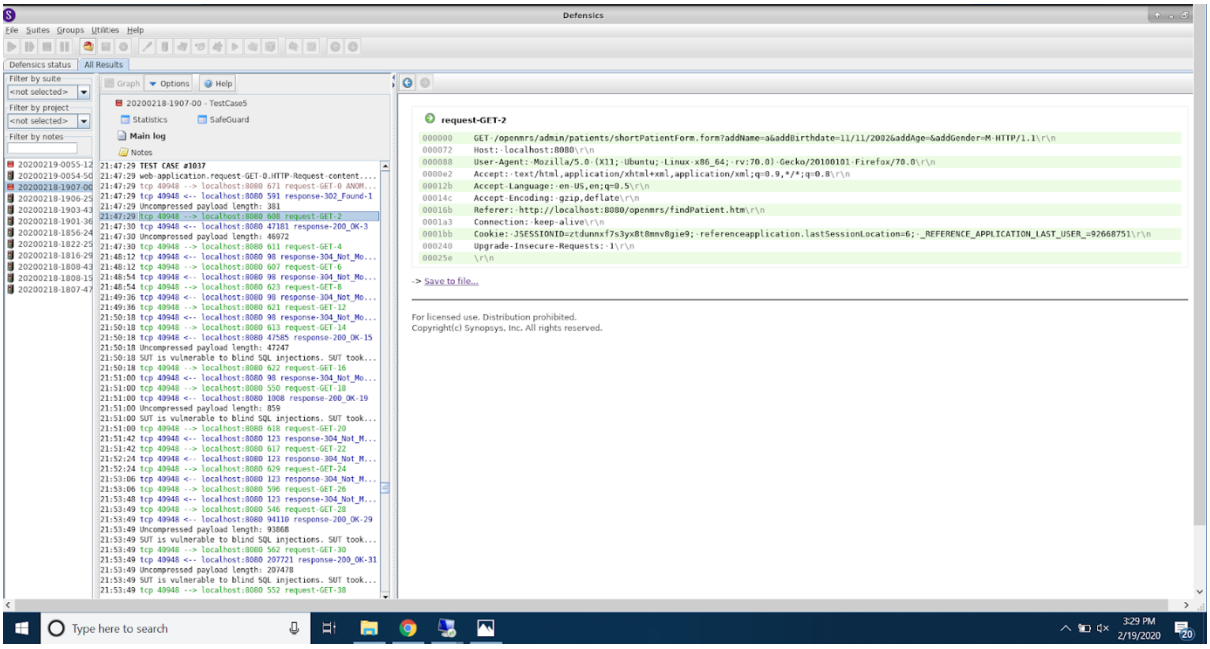
1. Navigate to the login page at http://localhost:8080/openmrs
2. Login as the admin (Default credentials are username: admin, password: Admin123), with "Inpatient Ward" selected as the location for this session.
3. Go to the url /openmrs/admin/patients/shortPatientForm.form?addName=a&addBirthdate=11/11/2002&addAge=&addGender=M

Expected Results:

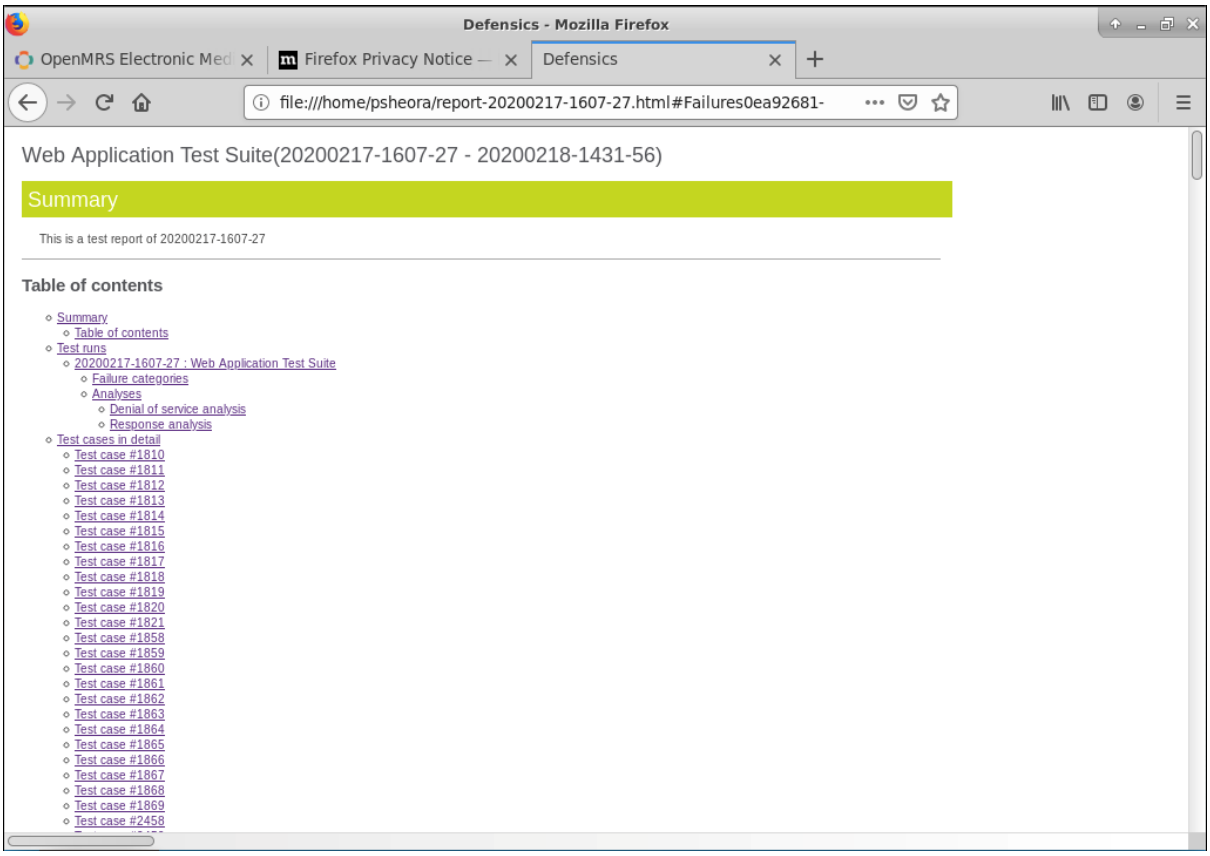
The user receives an error message.

Actual Results:

The user is directed to the add patient page with the form filled in with the information provided in the URL (Name: a, Birthdate: 11/11/2002, Gender: M).



Defensics Report Link: file:///home/psheora/report-20200217-1607-27.html



Test Case 2:

ASVS 14.3.1: Verify that web or application server and framework error messages are configured to deliver user actionable, customized responses to eliminate any unintended security disclosures.

CWE-209: Information Exposure Through an Error Message

Steps:

1.Navigate to the login page at http://localhost:8080/openmrs

2.Login as the admin (Default credentials are username: admin, password: Admin123), with "Inpatient Ward" selected as the location for this session.

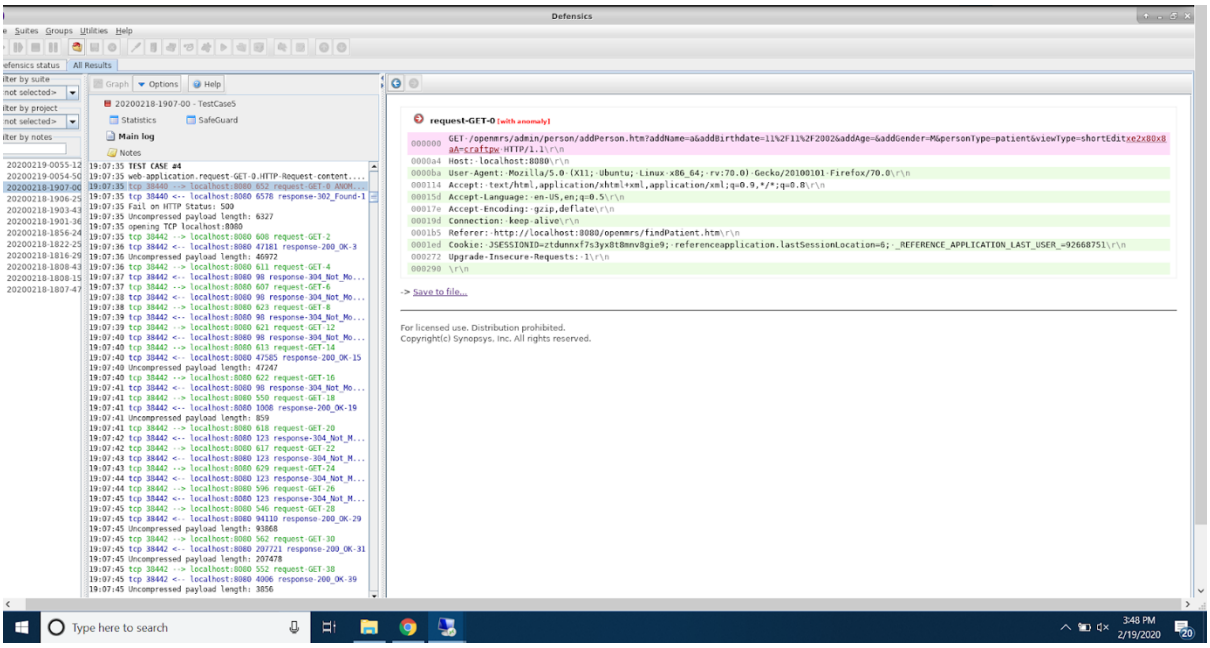
3.Go to the URL http://localhost:8080/openmrs/admin/person/addPerson.htm?addName=a&addBirthdate=11%2F11%2F2002&addAge=&addGender=M&personType=patient&viewType=shortEditxe2x80x8aA=craftpw

Expected Results:

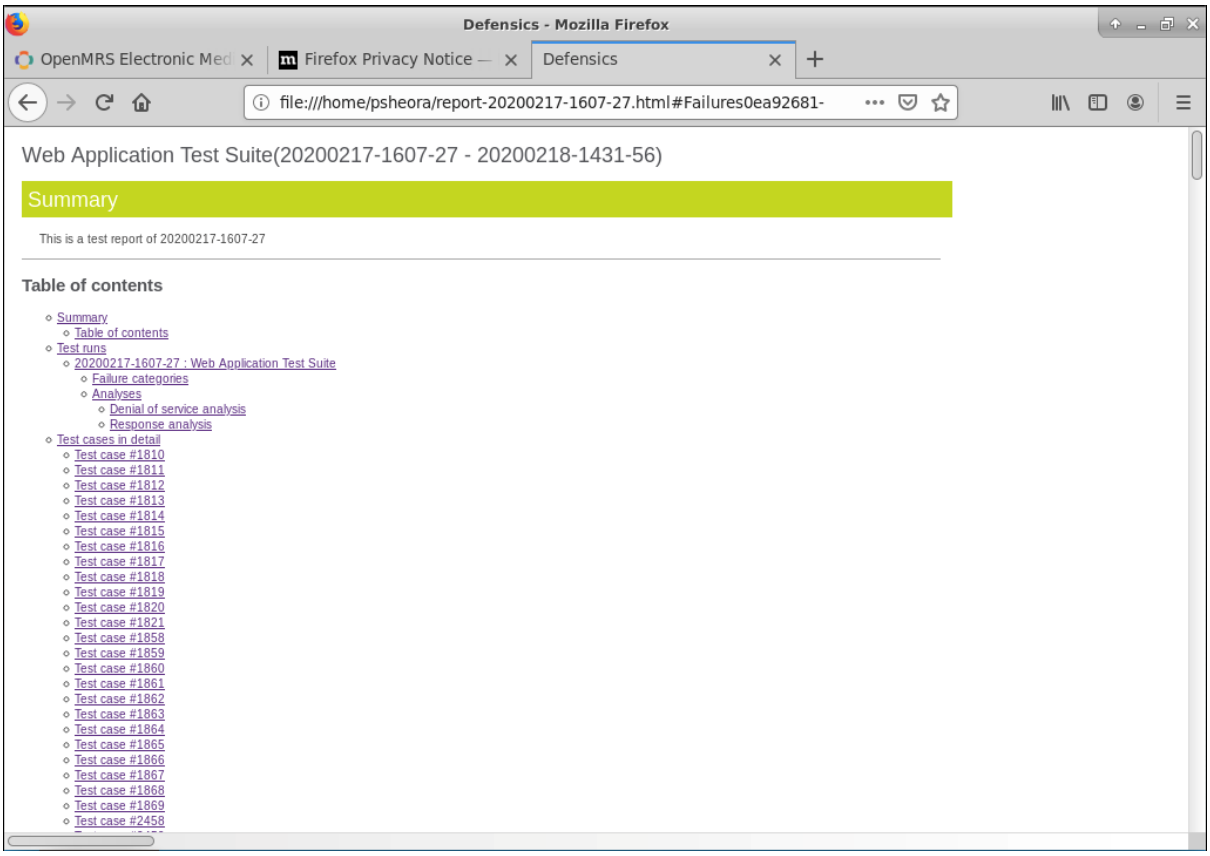
The user sees a generic error message, such as "Invalid Address."

Actual Results:

The user sees a detailed error message, including "javax.servlet.ServletException: You entered viewType = "shortEditxe2x80x8aA=craftpw" and personType = "patient" which is an invalid viewType/personType combination. Valid viewType/personType combinations are edit/patient, edit/user, shortEdit/patient, view/patient. The viewType edit is valid without any personType. Also, the personType user is valid without any viewType."



Defensics Report Link: file:///home/psheora/report-20200217-1607-27.html



Test Case 3:

ASVS 5.1.5: Verify that URL redirects and forwards only allow whitelisted destinations, or show a warning when redirecting to potentially untrusted content
CWE-601: URL Redirection to Untrusted Site('Open Redirect')

Steps:

1.Login as the admin using the username as 'admin', password as 'admin123' and location as "Inpatient Ward".

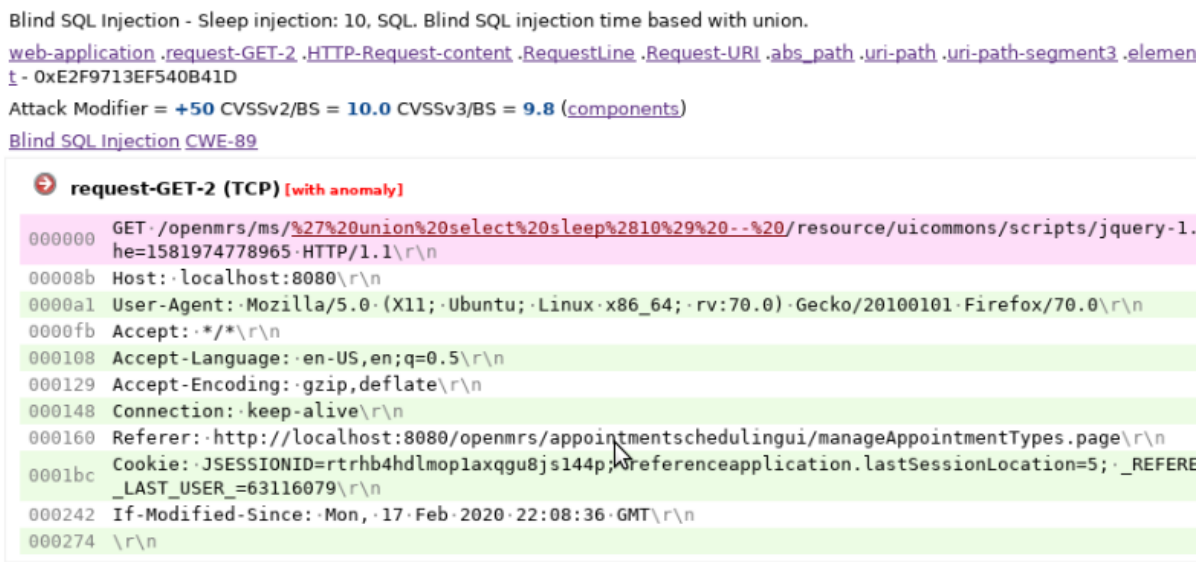
2.Go to the URL
https://localhost:8080/openmrs/ms/%27%20union%20select%20null%2Csleep%2810%29%20--%20/resource/uicommons/scripts/jquery-1.12.4.min.js?cache=1581974778965

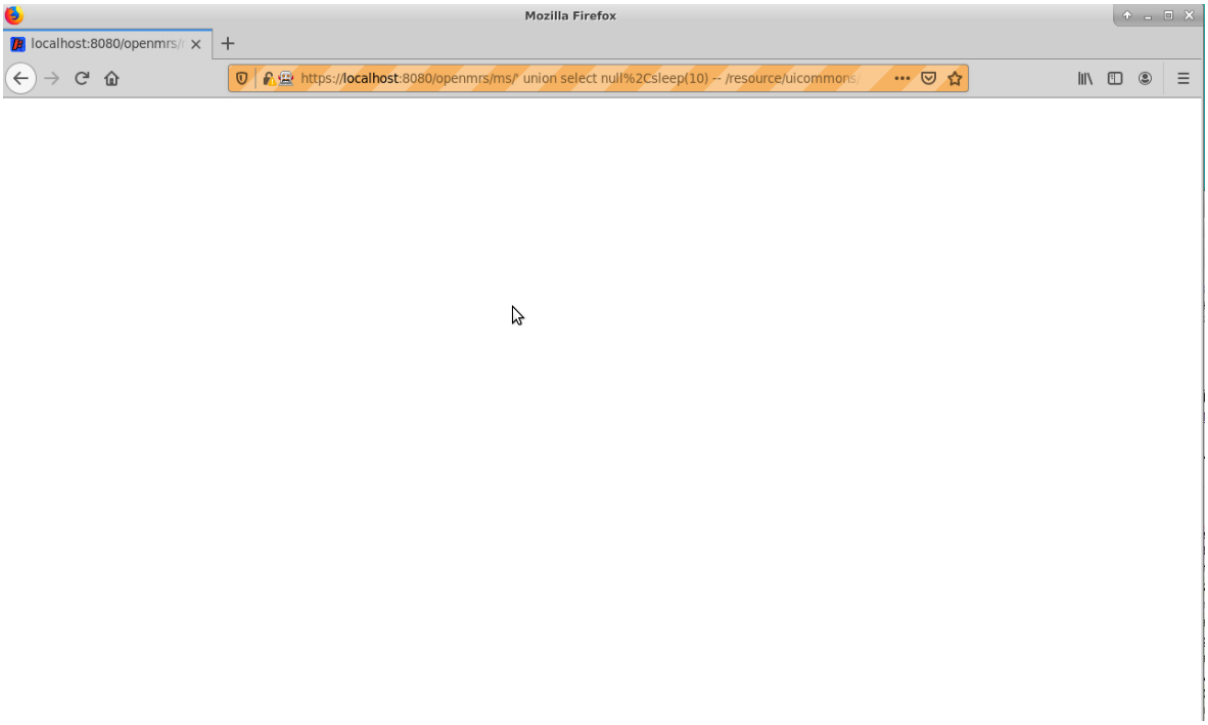
Expected Results:

The web application should throw a 404 error denoting a webpage is not present

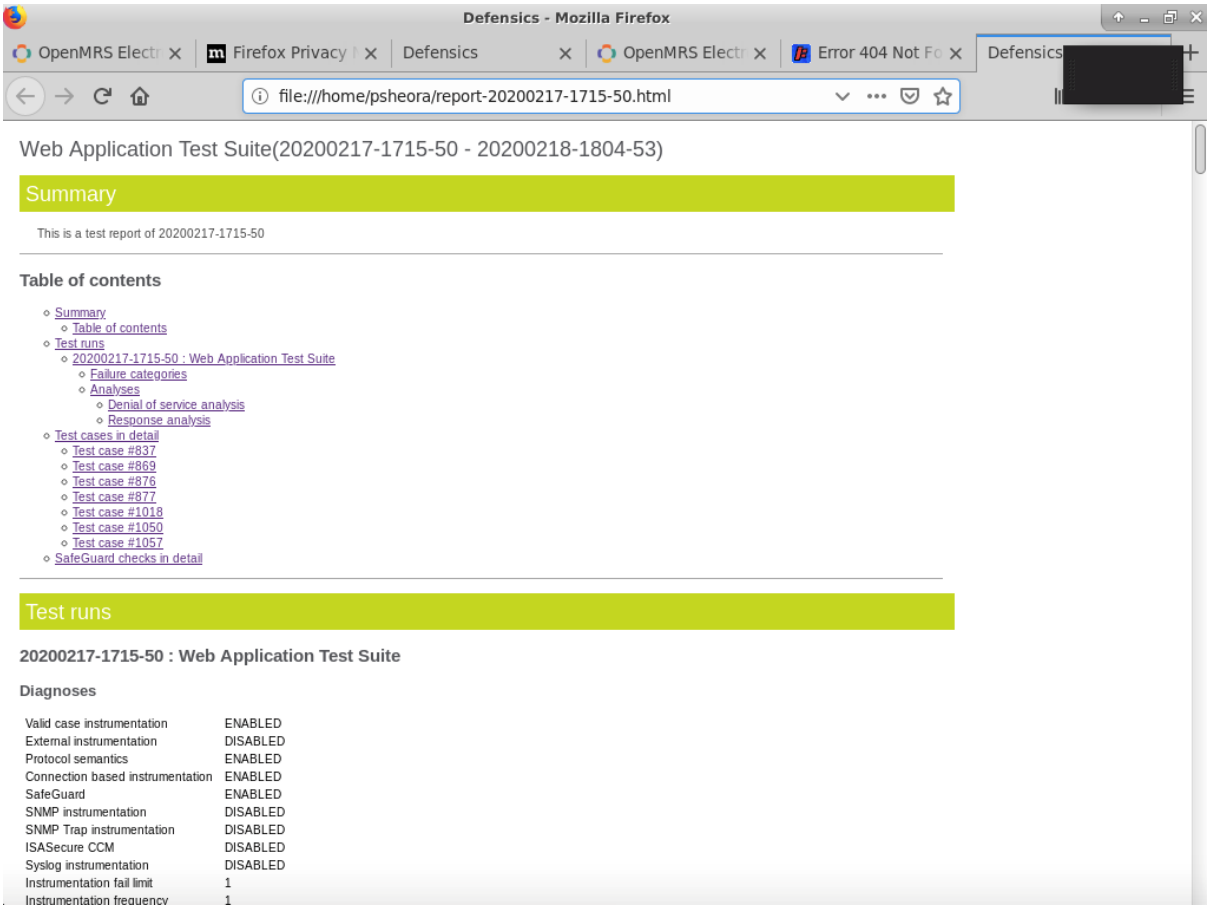
Actual Results:

The web application becomes inactive and we are not able to do anything about it.





Defensics Report Link: file:///home/psheora/report-20200217-1715-50.html



Test Case 4:

ASVS 12.3.1:Verify that user-submitted filename metadata is not used directly with system or framework file and URL API to protect against path traversal.

CWE-22: Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')

Steps:

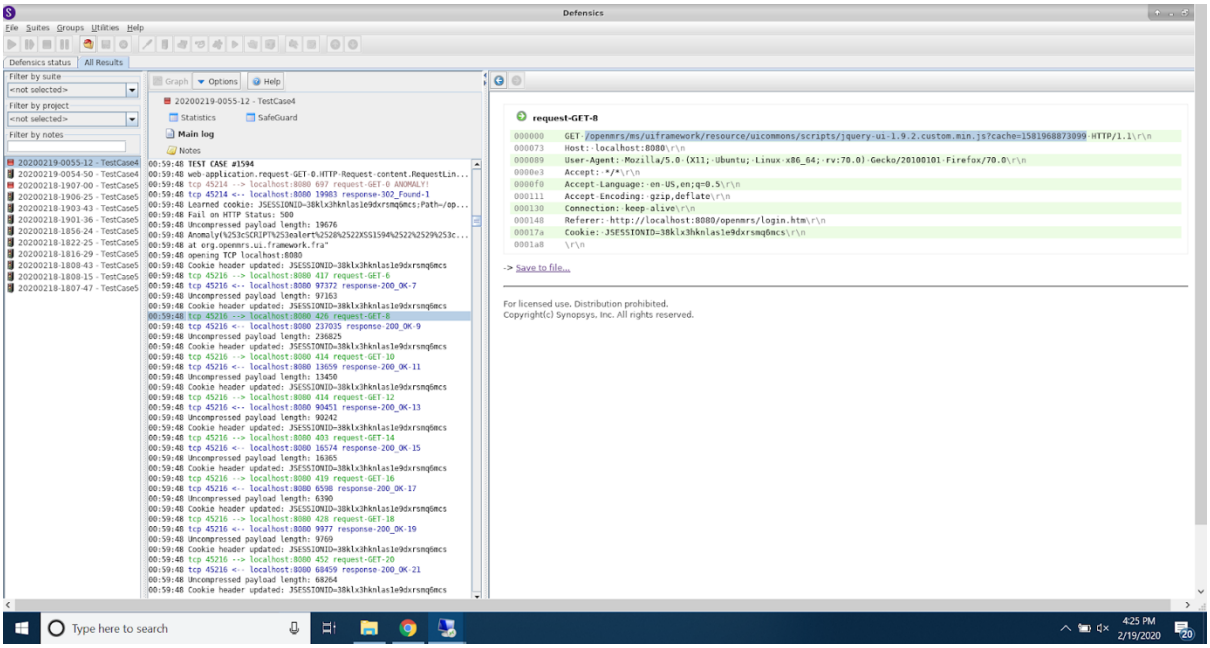
1. Go to the URL: http://localhost:8080/openmrs/ms/uiframework/resource/uicommons/scripts/jquery-ui-1.9.2.custom.min.js?cache=1581968873099

Expected Results:

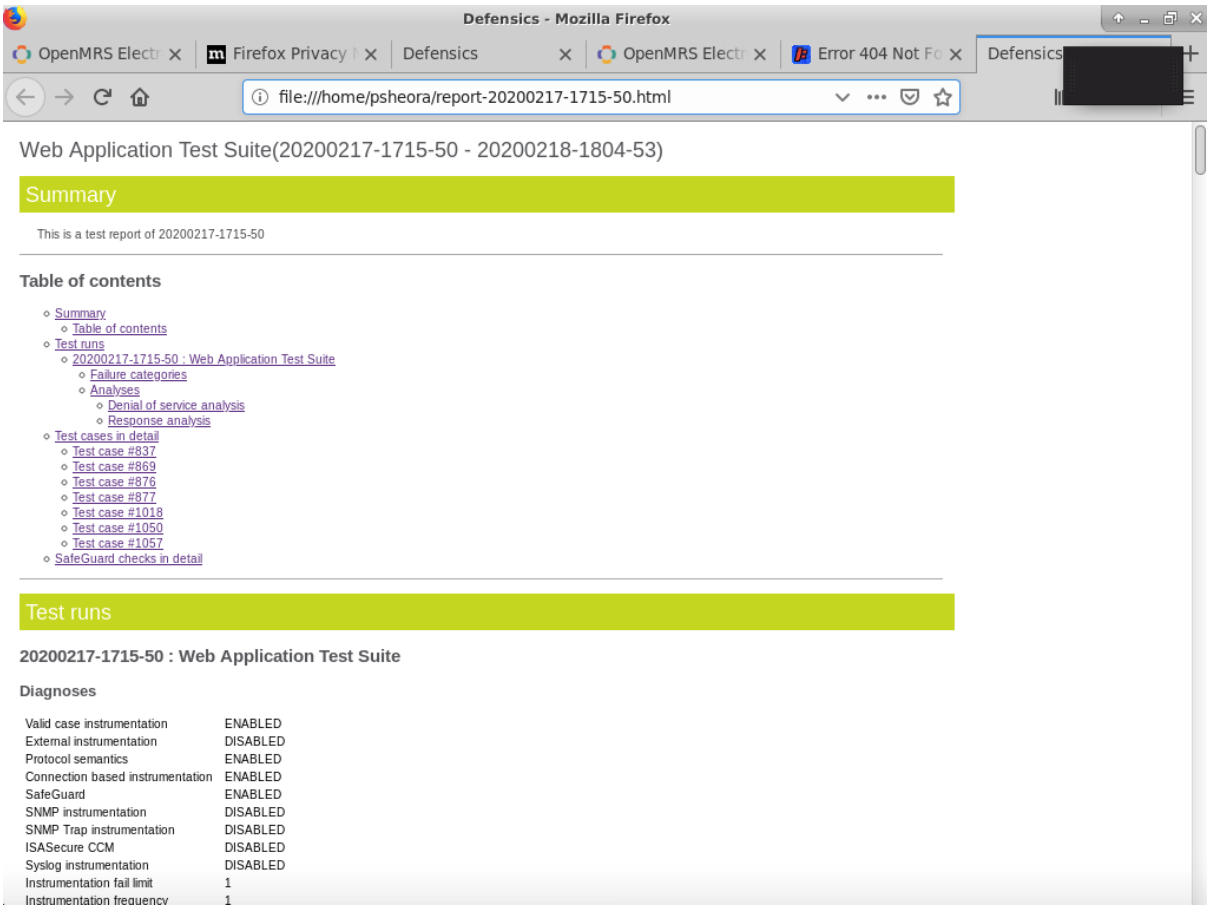
The user sees an error message, such as “Access not authorized.”

Actual Results:

The user can see the contents of the source file in their browser.



Defensics Report Link: file:///home/psheora/report-20200217-1715-50.html



Test Case 5:

ASVS 3.7.1: Verify the application ensures a valid login session or requires re-authentication or secondary verification before allowing any sensitive transactions or account modifications.

CWE-778: Insufficient Logging

Steps:

1.Login as the admin using the username as ‘admin’, password as ‘admin123’ and location as “Inpatient Ward”.

2.Log out of the account.

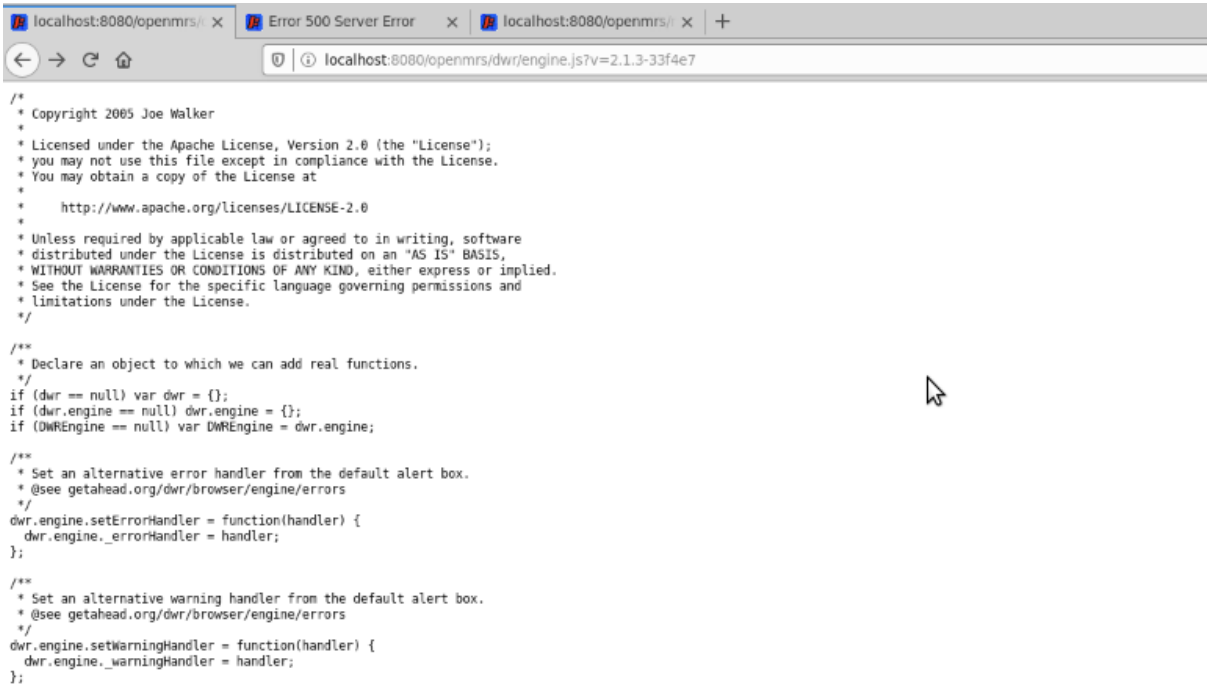
3.Go to the http://localhost:8080/openmrs/dwr/engine.js?v=2.1.3-33f4e7

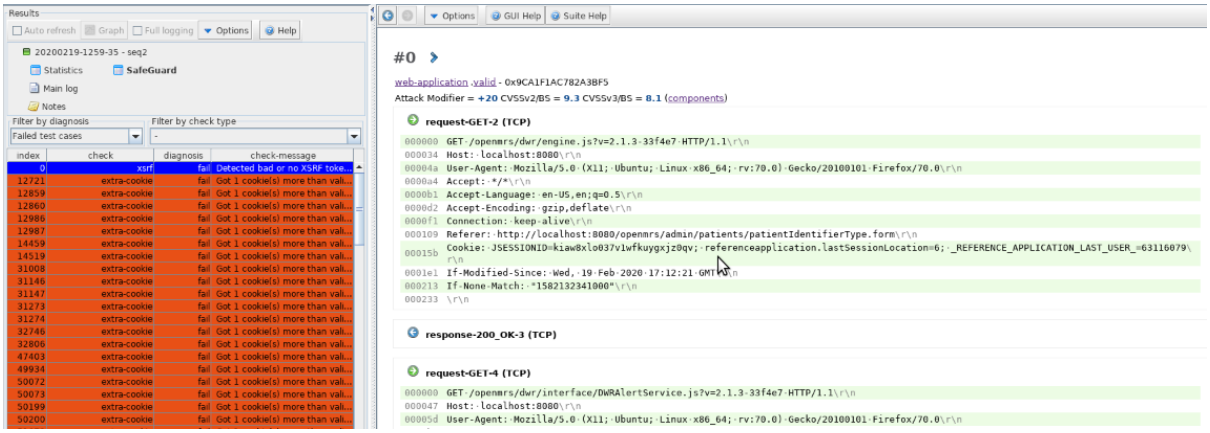
Expected Results:

The web application should not allow the user to access the code since the user is not logged in. The application should throw a suitable error and block the user from accessing it.

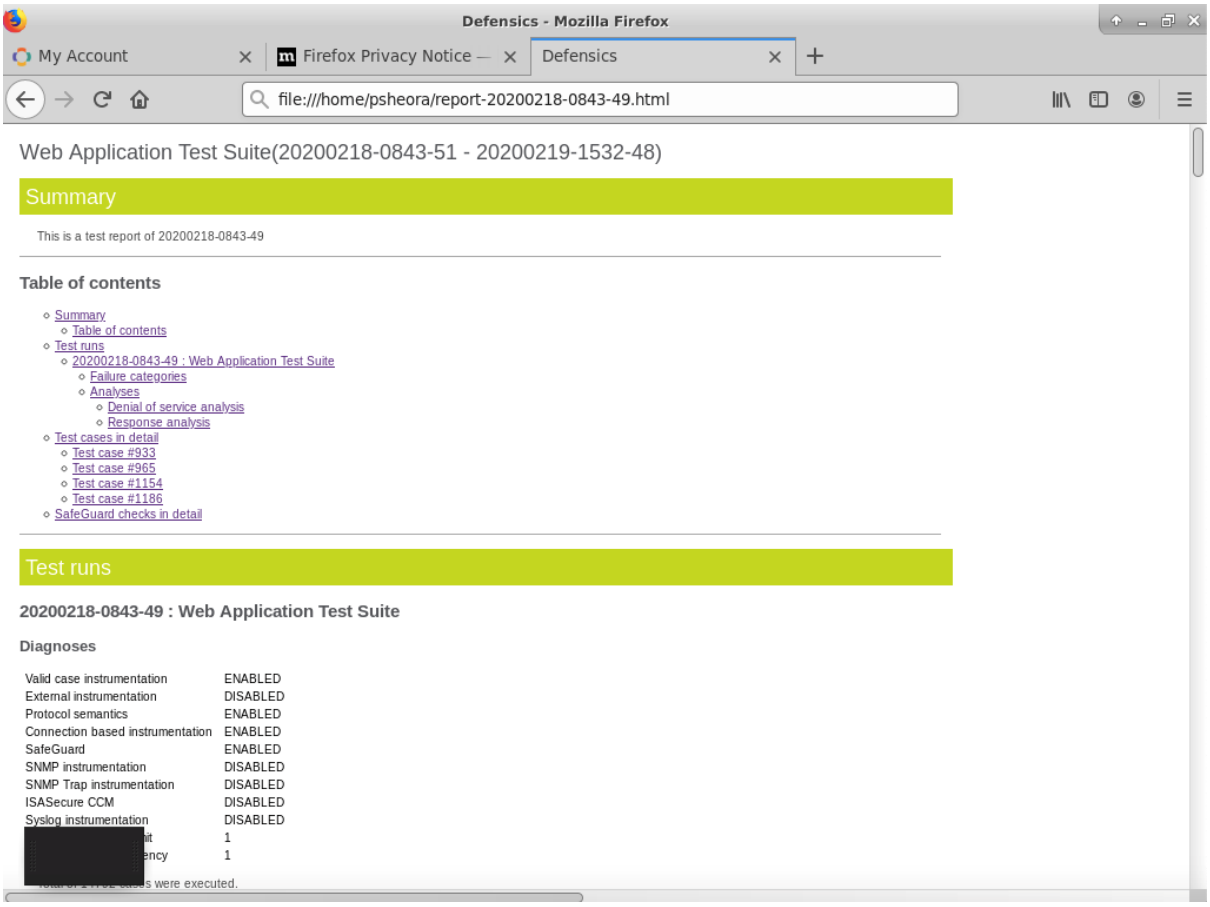
Actual Results:

The user is able to look at the source code





Defensics Report Link: file:///home/psheora/report-20200218-0843-49.html



Time Metrics for Defensics:
--- Total time to identify 5 vulnerabilities: 10 hours (about 1 vulnerability per 2 hour)
--- Total number of vulnerabilities found: 5 vulnerability

3. Vulnerable Dependencies

Task 1

Run all the five tools on all five modules of OpenMRS-core.

1. Report the results for each tool run on each module. The results should contain
   - The number of total vulnerable dependencies for the module

Ans. Number of vulnerabilities detected:

Tool	api module	tools module	test module	web module	webapp module
OWASP-Dependency-Check	135	0	35	200	293
RedHat Victims	13	0	3	21	21
GitHub's checker	–	–	–	–	–
Sonatype DepShield	–	–	–	–	–
Snyk	13	0	3	23	23

GitHub's checker vulnerabilities: 2 (Total)

Sonatype DepShield vulnerabilities: 15 (Total)

GitHub's Checker and Sonatype DepShield have reported vulnerabilities but they have not reported the module specific dependencies.

- The list of CVEs for each vulnerable dependency of the module

Ans.

a) OWASP-Dependency-Check

(1) Api module

DependencyName	CVE
mysql-connector-java-5.1.28.jar	CVE-2017-3523
mysql-connector-java-5.1.28.jar	CVE-2017-3589
mysql-connector-java-5.1.28.jar	CVE-2018-3258
mysql-connector-java-5.1.28.jar	CVE-2019-2692
postgresql-9.0-801.jdbc4.jar	CVE-2018-10936
postgresql-9.0-801.jdbc4.jar	CVE-2019-10210
postgresql-9.0-801.jdbc4.jar	CVE-2019-10211
commons-beanutils-1.7.0.jar	CVE-2014-0114
commons-beanutils-1.7.0.jar	CVE-2019-10086
log4j-1.2.15.jar	CVE-2019-17571
spring-core-4.1.4.RELEASE.jar	CVE-2015-0201
spring-core-4.1.4.RELEASE.jar	CVE-2015-3192
spring-core-4.1.4.RELEASE.jar	CVE-2015-5211
spring-core-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-core-4.1.4.RELEASE.jar	CVE-2016-5007
spring-core-4.1.4.RELEASE.jar	CVE-2018-1270
spring-core-4.1.4.RELEASE.jar	CVE-2018-1271
spring-core-4.1.4.RELEASE.jar	CVE-2018-1272
spring-beans-4.1.4.RELEASE.jar	CVE-2015-0201
spring-beans-4.1.4.RELEASE.jar	CVE-2015-3192
spring-beans-4.1.4.RELEASE.jar	CVE-2015-5211
spring-beans-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-beans-4.1.4.RELEASE.jar	CVE-2016-5007
spring-beans-4.1.4.RELEASE.jar	CVE-2018-1270
spring-beans-4.1.4.RELEASE.jar	CVE-2018-1271
spring-beans-4.1.4.RELEASE.jar	CVE-2018-1272
spring-context-4.1.4.RELEASE.jar	CVE-2015-0201
spring-context-4.1.4.RELEASE.jar	CVE-2015-3192
spring-context-4.1.4.RELEASE.jar	CVE-2015-5211
spring-context-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-context-4.1.4.RELEASE.jar	CVE-2016-5007
spring-context-4.1.4.RELEASE.jar	CVE-2018-1270
spring-context-4.1.4.RELEASE.jar	CVE-2018-1271
spring-context-4.1.4.RELEASE.jar	CVE-2018-1272
spring-expression-4.1.4.RELEASE.jar	CVE-2015-0201
spring-expression-4.1.4.RELEASE.jar	CVE-2015-3192
spring-expression-4.1.4.RELEASE.jar	CVE-2015-5211
spring-expression-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-expression-4.1.4.RELEASE.jar	CVE-2016-5007
spring-expression-4.1.4.RELEASE.jar	CVE-2018-1270
spring-expression-4.1.4.RELEASE.jar	CVE-2018-1271
spring-expression-4.1.4.RELEASE.jar	CVE-2018-1272

spring-aop-4.1.4.RELEASE.jar	CVE-2015-0201
spring-aop-4.1.4.RELEASE.jar	CVE-2015-3192
spring-aop-4.1.4.RELEASE.jar	CVE-2015-5211
spring-aop-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-aop-4.1.4.RELEASE.jar	CVE-2016-5007
spring-aop-4.1.4.RELEASE.jar	CVE-2018-1270
spring-aop-4.1.4.RELEASE.jar	CVE-2018-1271
spring-aop-4.1.4.RELEASE.jar	CVE-2018-1272
spring-orm-4.1.4.RELEASE.jar	CVE-2015-0201
spring-orm-4.1.4.RELEASE.jar	CVE-2015-3192
spring-orm-4.1.4.RELEASE.jar	CVE-2015-5211
spring-orm-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-orm-4.1.4.RELEASE.jar	CVE-2016-5007
spring-orm-4.1.4.RELEASE.jar	CVE-2018-1270
spring-orm-4.1.4.RELEASE.jar	CVE-2018-1271
spring-orm-4.1.4.RELEASE.jar	CVE-2018-1272
spring-tx-4.1.4.RELEASE.jar	CVE-2015-0201
spring-tx-4.1.4.RELEASE.jar	CVE-2015-3192
spring-tx-4.1.4.RELEASE.jar	CVE-2015-5211
spring-tx-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-tx-4.1.4.RELEASE.jar	CVE-2016-5007
spring-tx-4.1.4.RELEASE.jar	CVE-2018-1270
spring-tx-4.1.4.RELEASE.jar	CVE-2018-1271
spring-tx-4.1.4.RELEASE.jar	CVE-2018-1272
spring-jdbc-4.1.4.RELEASE.jar	CVE-2015-0201
spring-jdbc-4.1.4.RELEASE.jar	CVE-2015-3192
spring-jdbc-4.1.4.RELEASE.jar	CVE-2015-5211
spring-jdbc-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-jdbc-4.1.4.RELEASE.jar	CVE-2016-5007
spring-jdbc-4.1.4.RELEASE.jar	CVE-2018-1270
spring-jdbc-4.1.4.RELEASE.jar	CVE-2018-1271
spring-jdbc-4.1.4.RELEASE.jar	CVE-2018-1272
spring-context-support-4.1.4.RELEASE.jar	CVE-2015-0201
spring-context-support-4.1.4.RELEASE.jar	CVE-2015-3192
spring-context-support-4.1.4.RELEASE.jar	CVE-2015-5211
spring-context-support-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-context-support-4.1.4.RELEASE.jar	CVE-2016-5007
spring-context-support-4.1.4.RELEASE.jar	CVE-2018-1270

spring-context-support-4.1.4.RELEASE.jar	CVE-2018-1271
spring-context-support-4.1.4.RELEASE.jar	CVE-2018-1272
xalan-2.7.0.jar	CVE-2014-0107
c3p0-0.9.2.1.jar	CVE-2019-5427
dom4j-1.6.1.jar	CVE-2018-1000632
xstream-1.4.3.jar	CVE-2013-7285
xstream-1.4.3.jar	CVE-2016-3674
xstream-1.4.3.jar	CVE-2017-7957
modify-column-2.0.2.jar	CVE-2017-12796
modify-column-2.0.2.jar	CVE-2018-19276
identity-insert-1.2.1.jar	CVE-2017-12796
identity-insert-1.2.1.jar	CVE-2018-19276
type-converter-1.0.1.jar	CVE-2017-12796
type-converter-1.0.1.jar	CVE-2018-19276
xercesImpl-2.8.0.jar	CVE-2009-2625
xercesImpl-2.8.0.jar	CVE-2012-0881
hibernate-validator-4.2.0.Final.jar	CVE-2014-3558
jackson-mapper-asl-1.9.13.jar	CVE-2017-15095
jackson-mapper-asl-1.9.13.jar	CVE-2017-17485
jackson-mapper-asl-1.9.13.jar	CVE-2017-7525
jackson-mapper-asl-1.9.13.jar	CVE-2018-1000873
jackson-mapper-asl-1.9.13.jar	CVE-2018-14718
jackson-mapper-asl-1.9.13.jar	CVE-2018-5968
jackson-mapper-asl-1.9.13.jar	CVE-2018-7489
jackson-mapper-asl-1.9.13.jar	CVE-2019-10172
jackson-mapper-asl-1.9.13.jar	CVE-2019-14540
jackson-mapper-asl-1.9.13.jar	CVE-2019-16335
jackson-mapper-asl-1.9.13.jar	CVE-2019-17267
jackson-databind-2.9.0.jar	CVE-2017-15095
jackson-databind-2.9.0.jar	CVE-2018-1000873
jackson-databind-2.9.0.jar	CVE-2018-11307
jackson-databind-2.9.0.jar	CVE-2018-12022
jackson-databind-2.9.0.jar	CVE-2018-12023
jackson-databind-2.9.0.jar	CVE-2018-14718
jackson-databind-2.9.0.jar	CVE-2018-14719
jackson-databind-2.9.0.jar	CVE-2018-14720
jackson-databind-2.9.0.jar	CVE-2018-14721
jackson-databind-2.9.0.jar	CVE-2018-19360
jackson-databind-2.9.0.jar	CVE-2018-19361
jackson-databind-2.9.0.jar	CVE-2018-19362
jackson-databind-2.9.0.jar	CVE-2018-5968

jackson-databind-2.9.0.jar	CVE-2018-7489
jackson-databind-2.9.0.jar	CVE-2019-12086
jackson-databind-2.9.0.jar	CVE-2019-12384
jackson-databind-2.9.0.jar	CVE-2019-12814
jackson-databind-2.9.0.jar	CVE-2019-14379
jackson-databind-2.9.0.jar	CVE-2019-14439
jackson-databind-2.9.0.jar	CVE-2019-14540
jackson-databind-2.9.0.jar	CVE-2019-16335
jackson-databind-2.9.0.jar	CVE-2019-16942
jackson-databind-2.9.0.jar	CVE-2019-16943
jackson-databind-2.9.0.jar	CVE-2019-17267
jackson-databind-2.9.0.jar	CVE-2019-17531
jackson-databind-2.9.0.jar	CVE-2019-20330
groovy-all-2.4.6.jar	CVE-2016-6814

(2) Test module

DependencyName	CVE
spring-test-4.1.4.RELEASE.jar	CVE-2015-0201
spring-test-4.1.4.RELEASE.jar	CVE-2015-3192
spring-test-4.1.4.RELEASE.jar	CVE-2015-5211
spring-test-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-test-4.1.4.RELEASE.jar	CVE-2016-5007
spring-test-4.1.4.RELEASE.jar	CVE-2018-1270
spring-test-4.1.4.RELEASE.jar	CVE-2018-1271
spring-test-4.1.4.RELEASE.jar	CVE-2018-1272
spring-core-4.1.4.RELEASE.jar	CVE-2015-0201
spring-core-4.1.4.RELEASE.jar	CVE-2015-3192
spring-core-4.1.4.RELEASE.jar	CVE-2015-5211
spring-core-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-core-4.1.4.RELEASE.jar	CVE-2016-5007
spring-core-4.1.4.RELEASE.jar	CVE-2018-1270
spring-core-4.1.4.RELEASE.jar	CVE-2018-1271
spring-core-4.1.4.RELEASE.jar	CVE-2018-1272
poi-3.5-beta5.jar	CVE-2012-0213
poi-3.5-beta5.jar	CVE-2014-3529
poi-3.5-beta5.jar	CVE-2014-3574
poi-3.5-beta5.jar	CVE-2014-9527
poi-3.5-beta5.jar	CVE-2016-5000
poi-3.5-beta5.jar	CVE-2017-12626
poi-3.5-beta5.jar	CVE-2017-5644
poi-3.5-beta5.jar	CVE-2019-12415
log4j-1.2.15.jar	CVE-2019-17571

mysql-connector-java-5.1.28.jar	CVE-2017-3523
mysql-connector-java-5.1.28.jar	CVE-2017-3589
mysql-connector-java-5.1.28.jar	CVE-2018-3258
mysql-connector-java-5.1.28.jar	CVE-2019-2692
derbyclient-10.4.2.0.jar	CVE-2009-4269
derbyclient-10.4.2.0.jar	CVE-2015-1832
derbyclient-10.4.2.0.jar	CVE-2018-1313
postgresql-9.0-801.jdbc4.jar	CVE-2018-10936
postgresql-9.0-801.jdbc4.jar	CVE-2019-10210
postgresql-9.0-801.jdbc4.jar	CVE-2019-10211

(3) Tools module

0 vulnerabilities detected.

(4) Web module

DependencyName	CVE
openmrs-api-2.1.3.jar	CVE-2017-12796
openmrs-api-2.1.3.jar	CVE-2018-19276
commons-beanutils-1.7.0.jar	CVE-2014-0114
commons-beanutils-1.7.0.jar	CVE-2019-10086
log4j-1.2.15.jar	CVE-2019-17571
spring-core-4.1.4.RELEASE.jar	CVE-2015-0201
spring-core-4.1.4.RELEASE.jar	CVE-2015-3192
spring-core-4.1.4.RELEASE.jar	CVE-2015-5211
spring-core-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-core-4.1.4.RELEASE.jar	CVE-2016-5007
spring-core-4.1.4.RELEASE.jar	CVE-2018-1270
spring-core-4.1.4.RELEASE.jar	CVE-2018-1271
spring-core-4.1.4.RELEASE.jar	CVE-2018-1272
spring-beans-4.1.4.RELEASE.jar	CVE-2015-0201
spring-beans-4.1.4.RELEASE.jar	CVE-2015-3192
spring-beans-4.1.4.RELEASE.jar	CVE-2015-5211
spring-beans-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-beans-4.1.4.RELEASE.jar	CVE-2016-5007
spring-beans-4.1.4.RELEASE.jar	CVE-2018-1270
spring-beans-4.1.4.RELEASE.jar	CVE-2018-1271
spring-beans-4.1.4.RELEASE.jar	CVE-2018-1272
spring-context-4.1.4.RELEASE.jar	CVE-2015-0201
spring-context-4.1.4.RELEASE.jar	CVE-2015-3192
spring-context-4.1.4.RELEASE.jar	CVE-2015-5211
spring-context-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-context-4.1.4.RELEASE.jar	CVE-2016-5007

spring-context-4.1.4.RELEASE.jar	CVE-2018-1270
spring-context-4.1.4.RELEASE.jar	CVE-2018-1271
spring-context-4.1.4.RELEASE.jar	CVE-2018-1272
spring-aop-4.1.4.RELEASE.jar	CVE-2015-0201
spring-aop-4.1.4.RELEASE.jar	CVE-2015-3192
spring-aop-4.1.4.RELEASE.jar	CVE-2015-5211
spring-aop-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-aop-4.1.4.RELEASE.jar	CVE-2016-5007
spring-aop-4.1.4.RELEASE.jar	CVE-2018-1270
spring-aop-4.1.4.RELEASE.jar	CVE-2018-1271
spring-aop-4.1.4.RELEASE.jar	CVE-2018-1272
spring-orm-4.1.4.RELEASE.jar	CVE-2015-0201
spring-orm-4.1.4.RELEASE.jar	CVE-2015-3192
spring-orm-4.1.4.RELEASE.jar	CVE-2015-5211
spring-orm-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-orm-4.1.4.RELEASE.jar	CVE-2016-5007
spring-orm-4.1.4.RELEASE.jar	CVE-2018-1270
spring-orm-4.1.4.RELEASE.jar	CVE-2018-1271
spring-orm-4.1.4.RELEASE.jar	CVE-2018-1272
spring-tx-4.1.4.RELEASE.jar	CVE-2015-0201
spring-tx-4.1.4.RELEASE.jar	CVE-2015-3192
spring-tx-4.1.4.RELEASE.jar	CVE-2015-5211
spring-tx-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-tx-4.1.4.RELEASE.jar	CVE-2016-5007
spring-tx-4.1.4.RELEASE.jar	CVE-2018-1270
spring-tx-4.1.4.RELEASE.jar	CVE-2018-1271
spring-tx-4.1.4.RELEASE.jar	CVE-2018-1272
spring-jdbc-4.1.4.RELEASE.jar	CVE-2015-0201
spring-jdbc-4.1.4.RELEASE.jar	CVE-2015-3192
spring-jdbc-4.1.4.RELEASE.jar	CVE-2015-5211
spring-jdbc-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-jdbc-4.1.4.RELEASE.jar	CVE-2016-5007
spring-jdbc-4.1.4.RELEASE.jar	CVE-2018-1270
spring-jdbc-4.1.4.RELEASE.jar	CVE-2018-1271
spring-jdbc-4.1.4.RELEASE.jar	CVE-2018-1272
spring-context-support-4.1.4.RELEASE.jar	CVE-2015-0201
spring-context-support-4.1.4.RELEASE.jar	CVE-2015-3192
spring-context-support-4.1.4.RELEASE.jar	CVE-2015-5211
spring-context-support-4.1.4.RELEASE.jar	CVE-2016-1000027

spring-context-support-4.1.4.RELEASE.jar	CVE-2016-5007
spring-context-support-4.1.4.RELEASE.jar	CVE-2018-1270
spring-context-support-4.1.4.RELEASE.jar	CVE-2018-1271
spring-context-support-4.1.4.RELEASE.jar	CVE-2018-1272
xalan-2.7.0.jar	CVE-2014-0107
c3p0-0.9.2.1.jar	CVE-2019-5427
dom4j-1.6.1.jar	CVE-2018-1000632
xstream-1.4.3.jar	CVE-2013-7285
xstream-1.4.3.jar	CVE-2016-3674
xstream-1.4.3.jar	CVE-2017-7957
modify-column-2.0.2.jar	CVE-2017-12796
modify-column-2.0.2.jar	CVE-2018-19276
identity-insert-1.2.1.jar	CVE-2017-12796
identity-insert-1.2.1.jar	CVE-2018-19276
type-converter-1.0.1.jar	CVE-2017-12796
type-converter-1.0.1.jar	CVE-2018-19276
xercesImpl-2.8.0.jar	CVE-2009-2625
xercesImpl-2.8.0.jar	CVE-2012-0881
hibernate-validator-4.2.0.Final.jar	CVE-2014-3558
jackson-mapper-asl-1.9.13.jar	CVE-2017-15095
jackson-mapper-asl-1.9.13.jar	CVE-2017-17485
jackson-mapper-asl-1.9.13.jar	CVE-2017-7525
jackson-mapper-asl-1.9.13.jar	CVE-2018-1000873
jackson-mapper-asl-1.9.13.jar	CVE-2018-14718
jackson-mapper-asl-1.9.13.jar	CVE-2018-5968
jackson-mapper-asl-1.9.13.jar	CVE-2018-7489
jackson-mapper-asl-1.9.13.jar	CVE-2019-10172
jackson-mapper-asl-1.9.13.jar	CVE-2019-14540
jackson-mapper-asl-1.9.13.jar	CVE-2019-16335
jackson-mapper-asl-1.9.13.jar	CVE-2019-17267
jackson-databind-2.9.0.jar	CVE-2017-15095
jackson-databind-2.9.0.jar	CVE-2018-1000873
jackson-databind-2.9.0.jar	CVE-2018-11307
jackson-databind-2.9.0.jar	CVE-2018-12022
jackson-databind-2.9.0.jar	CVE-2018-12023
jackson-databind-2.9.0.jar	CVE-2018-14718
jackson-databind-2.9.0.jar	CVE-2018-14719
jackson-databind-2.9.0.jar	CVE-2018-14720
jackson-databind-2.9.0.jar	CVE-2018-14721
jackson-databind-2.9.0.jar	CVE-2018-19360

jackson-databind-2.9.0.jar	CVE-2018-19361
jackson-databind-2.9.0.jar	CVE-2018-19362
jackson-databind-2.9.0.jar	CVE-2018-5968
jackson-databind-2.9.0.jar	CVE-2018-7489
jackson-databind-2.9.0.jar	CVE-2019-12086
jackson-databind-2.9.0.jar	CVE-2019-12384
jackson-databind-2.9.0.jar	CVE-2019-12814
jackson-databind-2.9.0.jar	CVE-2019-14379
jackson-databind-2.9.0.jar	CVE-2019-14439
jackson-databind-2.9.0.jar	CVE-2019-14540
jackson-databind-2.9.0.jar	CVE-2019-16335
jackson-databind-2.9.0.jar	CVE-2019-16942
jackson-databind-2.9.0.jar	CVE-2019-16943
jackson-databind-2.9.0.jar	CVE-2019-17267
jackson-databind-2.9.0.jar	CVE-2019-17531
jackson-databind-2.9.0.jar	CVE-2019-20330
groovy-all-2.4.6.jar	CVE-2016-6814
jstl-1.1.2.jar	CVE-2015-0254
commons-fileupload-1.2.1.jar	Arbitrary file upload via deserialization
commons-fileupload-1.2.1.jar	CVE-2013-0248
commons-fileupload-1.2.1.jar	CVE-2014-0050
commons-fileupload-1.2.1.jar	CVE-2016-1000031
commons-fileupload-1.2.1.jar	CVE-2016-3092
spring-web-4.1.4.RELEASE.jar	CVE-2015-0201
spring-web-4.1.4.RELEASE.jar	CVE-2015-3192
spring-web-4.1.4.RELEASE.jar	CVE-2015-5211
spring-web-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-web-4.1.4.RELEASE.jar	CVE-2016-5007
spring-web-4.1.4.RELEASE.jar	CVE-2018-1270
spring-web-4.1.4.RELEASE.jar	CVE-2018-1271
spring-web-4.1.4.RELEASE.jar	CVE-2018-1272
spring-webmvc-4.1.4.RELEASE.jar	CVE-2015-0201
spring-webmvc-4.1.4.RELEASE.jar	CVE-2015-3192
spring-webmvc-4.1.4.RELEASE.jar	CVE-2015-5211
spring-webmvc-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-webmvc-4.1.4.RELEASE.jar	CVE-2016-5007
spring-webmvc-4.1.4.RELEASE.jar	CVE-2018-1270
spring-webmvc-4.1.4.RELEASE.jar	CVE-2018-1271
spring-webmvc-4.1.4.RELEASE.jar	CVE-2018-1272
spring-expression-4.1.4.RELEASE.jar	CVE-2015-0201
spring-expression-4.1.4.RELEASE.jar	CVE-2015-3192
spring-expression-4.1.4.RELEASE.jar	CVE-2015-5211

spring-expression-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-expression-4.1.4.RELEASE.jar	CVE-2016-5007
spring-expression-4.1.4.RELEASE.jar	CVE-2018-1270
spring-expression-4.1.4.RELEASE.jar	CVE-2018-1271
spring-expression-4.1.4.RELEASE.jar	CVE-2018-1272
spring-oxm-4.1.4.RELEASE.jar	CVE-2015-0201
spring-oxm-4.1.4.RELEASE.jar	CVE-2015-3192
spring-oxm-4.1.4.RELEASE.jar	CVE-2015-5211
spring-oxm-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-oxm-4.1.4.RELEASE.jar	CVE-2016-5007
spring-oxm-4.1.4.RELEASE.jar	CVE-2018-1270
spring-oxm-4.1.4.RELEASE.jar	CVE-2018-1271
spring-oxm-4.1.4.RELEASE.jar	CVE-2018-1272
mysql-connector-java-5.1.28.jar	CVE-2017-3523
mysql-connector-java-5.1.28.jar	CVE-2017-3589
mysql-connector-java-5.1.28.jar	CVE-2018-3258
mysql-connector-java-5.1.28.jar	CVE-2019-2692
postgresql-9.0-801.jdbc4.jar	CVE-2018-10936
postgresql-9.0-801.jdbc4.jar	CVE-2019-10210
postgresql-9.0-801.jdbc4.jar	CVE-2019-10211
standard-1.1.2.jar	CVE-2015-0254
struts-core-1.3.8.jar	CVE-2011-5057
struts-core-1.3.8.jar	CVE-2012-0391
struts-core-1.3.8.jar	CVE-2012-0392
struts-core-1.3.8.jar	CVE-2012-0393
struts-core-1.3.8.jar	CVE-2012-0394
struts-core-1.3.8.jar	CVE-2012-0838
struts-core-1.3.8.jar	CVE-2013-1965
struts-core-1.3.8.jar	CVE-2013-1966
struts-core-1.3.8.jar	CVE-2013-2115
struts-core-1.3.8.jar	CVE-2013-2134
struts-core-1.3.8.jar	CVE-2013-2135
struts-core-1.3.8.jar	CVE-2014-0094
struts-core-1.3.8.jar	CVE-2014-0113
struts-core-1.3.8.jar	CVE-2014-0114
struts-core-1.3.8.jar	CVE-2015-0899
struts-core-1.3.8.jar	CVE-2015-5169
struts-core-1.3.8.jar	CVE-2016-0785
struts-core-1.3.8.jar	CVE-2016-1181
struts-core-1.3.8.jar	CVE-2016-1182
struts-core-1.3.8.jar	CVE-2016-4003
struts-taglib-1.3.8.jar	CVE-2012-0394
struts-taglib-1.3.8.jar	CVE-2013-2115

struts-taglib-1.3.8.jar	CVE-2014-0114
struts-taglib-1.3.8.jar	CVE-2015-0899
struts-taglib-1.3.8.jar	CVE-2016-1181
struts-taglib-1.3.8.jar	CVE-2016-1182
struts-tiles-1.3.8.jar	CVE-2012-0394
struts-tiles-1.3.8.jar	CVE-2013-2115
struts-tiles-1.3.8.jar	CVE-2014-0114
struts-tiles-1.3.8.jar	CVE-2015-0899
struts-tiles-1.3.8.jar	CVE-2016-1181
struts-tiles-1.3.8.jar	CVE-2016-1182

(5) Webapp module

openmrs-api-2.1.3.jar	CVE-2017-12796
openmrs-api-2.1.3.jar	CVE-2018-19276
commons-beanutils-1.7.0.jar	CVE-2014-0114
commons-beanutils-1.7.0.jar	CVE-2019-10086
log4j-1.2.15.jar	CVE-2019-17571
spring-core-4.1.4.RELEASE.jar	CVE-2015-0201
spring-core-4.1.4.RELEASE.jar	CVE-2015-3192
spring-core-4.1.4.RELEASE.jar	CVE-2015-5211
spring-core-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-core-4.1.4.RELEASE.jar	CVE-2016-5007
spring-core-4.1.4.RELEASE.jar	CVE-2018-1270
spring-core-4.1.4.RELEASE.jar	CVE-2018-1271
spring-core-4.1.4.RELEASE.jar	CVE-2018-1272
spring-beans-4.1.4.RELEASE.jar	CVE-2015-0201
spring-beans-4.1.4.RELEASE.jar	CVE-2015-3192
spring-beans-4.1.4.RELEASE.jar	CVE-2015-5211
spring-beans-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-beans-4.1.4.RELEASE.jar	CVE-2016-5007
spring-beans-4.1.4.RELEASE.jar	CVE-2018-1270
spring-beans-4.1.4.RELEASE.jar	CVE-2018-1271
spring-beans-4.1.4.RELEASE.jar	CVE-2018-1272
spring-context-4.1.4.RELEASE.jar	CVE-2015-0201
spring-context-4.1.4.RELEASE.jar	CVE-2015-3192
spring-context-4.1.4.RELEASE.jar	CVE-2015-5211
spring-context-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-context-4.1.4.RELEASE.jar	CVE-2016-5007
spring-context-4.1.4.RELEASE.jar	CVE-2018-1270
spring-context-4.1.4.RELEASE.jar	CVE-2018-1271
spring-context-4.1.4.RELEASE.jar	CVE-2018-1272

spring-expression-4.1.4.RELEASE.jar	CVE-2015-0201
spring-expression-4.1.4.RELEASE.jar	CVE-2015-3192
spring-expression-4.1.4.RELEASE.jar	CVE-2015-5211
spring-expression-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-expression-4.1.4.RELEASE.jar	CVE-2016-5007
spring-expression-4.1.4.RELEASE.jar	CVE-2018-1270
spring-expression-4.1.4.RELEASE.jar	CVE-2018-1271
spring-expression-4.1.4.RELEASE.jar	CVE-2018-1272
spring-aop-4.1.4.RELEASE.jar	CVE-2015-0201
spring-aop-4.1.4.RELEASE.jar	CVE-2015-3192
spring-aop-4.1.4.RELEASE.jar	CVE-2015-5211
spring-aop-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-aop-4.1.4.RELEASE.jar	CVE-2016-5007
spring-aop-4.1.4.RELEASE.jar	CVE-2018-1270
spring-aop-4.1.4.RELEASE.jar	CVE-2018-1271
spring-aop-4.1.4.RELEASE.jar	CVE-2018-1272
spring-orm-4.1.4.RELEASE.jar	CVE-2015-0201
spring-orm-4.1.4.RELEASE.jar	CVE-2015-3192
spring-orm-4.1.4.RELEASE.jar	CVE-2015-5211
spring-orm-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-orm-4.1.4.RELEASE.jar	CVE-2016-5007
spring-orm-4.1.4.RELEASE.jar	CVE-2018-1270
spring-orm-4.1.4.RELEASE.jar	CVE-2018-1271
spring-orm-4.1.4.RELEASE.jar	CVE-2018-1272
spring-tx-4.1.4.RELEASE.jar	CVE-2015-0201
spring-tx-4.1.4.RELEASE.jar	CVE-2015-3192
spring-tx-4.1.4.RELEASE.jar	CVE-2015-5211
spring-tx-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-tx-4.1.4.RELEASE.jar	CVE-2016-5007
spring-tx-4.1.4.RELEASE.jar	CVE-2018-1270
spring-tx-4.1.4.RELEASE.jar	CVE-2018-1271
spring-tx-4.1.4.RELEASE.jar	CVE-2018-1272
spring-jdbc-4.1.4.RELEASE.jar	CVE-2015-0201
spring-jdbc-4.1.4.RELEASE.jar	CVE-2015-3192
spring-jdbc-4.1.4.RELEASE.jar	CVE-2015-5211
spring-jdbc-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-jdbc-4.1.4.RELEASE.jar	CVE-2016-5007
spring-jdbc-4.1.4.RELEASE.jar	CVE-2018-1270
spring-jdbc-4.1.4.RELEASE.jar	CVE-2018-1271
spring-jdbc-4.1.4.RELEASE.jar	CVE-2018-1272
spring-context-support-4.1.4.RELEASE.jar	CVE-2015-0201

spring-context-support-4.1.4.RELEASE.jar	CVE-2015-3192
spring-context-support-4.1.4.RELEASE.jar	CVE-2015-5211
spring-context-support-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-context-support-4.1.4.RELEASE.jar	CVE-2016-5007
spring-context-support-4.1.4.RELEASE.jar	CVE-2018-1270
spring-context-support-4.1.4.RELEASE.jar	CVE-2018-1271
spring-context-support-4.1.4.RELEASE.jar	CVE-2018-1272
xalan-2.7.0.jar	CVE-2014-0107
c3p0-0.9.2.1.jar	CVE-2019-5427
dom4j-1.6.1.jar	CVE-2018-1000632
xstream-1.4.3.jar	CVE-2013-7285
xstream-1.4.3.jar	CVE-2016-3674
xstream-1.4.3.jar	CVE-2017-7957
modify-column-2.0.2.jar	CVE-2017-12796
modify-column-2.0.2.jar	CVE-2018-19276
identity-insert-1.2.1.jar	CVE-2017-12796
identity-insert-1.2.1.jar	CVE-2018-19276
type-converter-1.0.1.jar	CVE-2017-12796
type-converter-1.0.1.jar	CVE-2018-19276
xercesImpl-2.8.0.jar	CVE-2009-2625
xercesImpl-2.8.0.jar	CVE-2012-0881
hibernate-validator-4.2.0.Final.jar	CVE-2014-3558
jackson-mapper-asl-1.9.13.jar	CVE-2017-15095
jackson-mapper-asl-1.9.13.jar	CVE-2017-17485
jackson-mapper-asl-1.9.13.jar	CVE-2017-7525
jackson-mapper-asl-1.9.13.jar	CVE-2018-1000873
jackson-mapper-asl-1.9.13.jar	CVE-2018-14718
jackson-mapper-asl-1.9.13.jar	CVE-2018-5968
jackson-mapper-asl-1.9.13.jar	CVE-2018-7489
jackson-mapper-asl-1.9.13.jar	CVE-2019-10172
jackson-mapper-asl-1.9.13.jar	CVE-2019-14540
jackson-mapper-asl-1.9.13.jar	CVE-2019-16335
jackson-mapper-asl-1.9.13.jar	CVE-2019-17267
jackson-databind-2.9.0.jar	CVE-2017-15095
jackson-databind-2.9.0.jar	CVE-2018-1000873
jackson-databind-2.9.0.jar	CVE-2018-11307
jackson-databind-2.9.0.jar	CVE-2018-12022

jackson-databind-2.9.0.jar	CVE-2018-12023
jackson-databind-2.9.0.jar	CVE-2018-14718
jackson-databind-2.9.0.jar	CVE-2018-14719
jackson-databind-2.9.0.jar	CVE-2018-14720
jackson-databind-2.9.0.jar	CVE-2018-14721
jackson-databind-2.9.0.jar	CVE-2018-19360
jackson-databind-2.9.0.jar	CVE-2018-19361
jackson-databind-2.9.0.jar	CVE-2018-19362
jackson-databind-2.9.0.jar	CVE-2018-5968
jackson-databind-2.9.0.jar	CVE-2018-7489
jackson-databind-2.9.0.jar	CVE-2019-12086
jackson-databind-2.9.0.jar	CVE-2019-12384
jackson-databind-2.9.0.jar	CVE-2019-12814
jackson-databind-2.9.0.jar	CVE-2019-14379
jackson-databind-2.9.0.jar	CVE-2019-14439
jackson-databind-2.9.0.jar	CVE-2019-14540
jackson-databind-2.9.0.jar	CVE-2019-16335
jackson-databind-2.9.0.jar	CVE-2019-16942
jackson-databind-2.9.0.jar	CVE-2019-16943
jackson-databind-2.9.0.jar	CVE-2019-17267
jackson-databind-2.9.0.jar	CVE-2019-17531
jackson-databind-2.9.0.jar	CVE-2019-20330
groovy-all-2.4.6.jar	CVE-2016-6814
openmrs-web-2.1.3.jar	CVE-2017-12796
openmrs-web-2.1.3.jar	CVE-2018-19276
jstl-1.1.2.jar	CVE-2015-0254
commons-fileupload-1.2.1.jar	Arbitrary file upload via deserialization
commons-fileupload-1.2.1.jar	CVE-2013-0248
commons-fileupload-1.2.1.jar	CVE-2014-0050
commons-fileupload-1.2.1.jar	CVE-2016-1000031
commons-fileupload-1.2.1.jar	CVE-2016-3092
spring-web-4.1.4.RELEASE.jar	CVE-2015-0201
spring-web-4.1.4.RELEASE.jar	CVE-2015-3192
spring-web-4.1.4.RELEASE.jar	CVE-2015-5211
spring-web-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-web-4.1.4.RELEASE.jar	CVE-2016-5007
spring-web-4.1.4.RELEASE.jar	CVE-2018-1270
spring-web-4.1.4.RELEASE.jar	CVE-2018-1271
spring-web-4.1.4.RELEASE.jar	CVE-2018-1272
spring-webmvc-4.1.4.RELEASE.jar	CVE-2015-0201
spring-webmvc-4.1.4.RELEASE.jar	CVE-2015-3192
spring-webmvc-4.1.4.RELEASE.jar	CVE-2015-5211
spring-webmvc-4.1.4.RELEASE.jar	CVE-2016-1000027

spring-webmvc-4.1.4.RELEASE.jar	CVE-2016-5007
spring-webmvc-4.1.4.RELEASE.jar	CVE-2018-1270
spring-webmvc-4.1.4.RELEASE.jar	CVE-2018-1271
spring-webmvc-4.1.4.RELEASE.jar	CVE-2018-1272
spring-oxm-4.1.4.RELEASE.jar	CVE-2015-0201
spring-oxm-4.1.4.RELEASE.jar	CVE-2015-3192
spring-oxm-4.1.4.RELEASE.jar	CVE-2015-5211
spring-oxm-4.1.4.RELEASE.jar	CVE-2016-1000027
spring-oxm-4.1.4.RELEASE.jar	CVE-2016-5007
spring-oxm-4.1.4.RELEASE.jar	CVE-2018-1270
spring-oxm-4.1.4.RELEASE.jar	CVE-2018-1271
spring-oxm-4.1.4.RELEASE.jar	CVE-2018-1272
mysql-connector-java-5.1.28.jar	CVE-2017-3523
mysql-connector-java-5.1.28.jar	CVE-2017-3589
mysql-connector-java-5.1.28.jar	CVE-2018-3258
mysql-connector-java-5.1.28.jar	CVE-2019-2692
postgresql-9.0-801.jdbc4.jar	CVE-2018-10936
postgresql-9.0-801.jdbc4.jar	CVE-2019-10210
postgresql-9.0-801.jdbc4.jar	CVE-2019-10211
standard-1.1.2.jar	CVE-2015-0254
struts-core-1.3.8.jar	CVE-2011-5057
struts-core-1.3.8.jar	CVE-2012-0391
struts-core-1.3.8.jar	CVE-2012-0392
struts-core-1.3.8.jar	CVE-2012-0393
struts-core-1.3.8.jar	CVE-2012-0394
struts-core-1.3.8.jar	CVE-2012-0838
struts-core-1.3.8.jar	CVE-2013-1965
struts-core-1.3.8.jar	CVE-2013-1966
struts-core-1.3.8.jar	CVE-2013-2115
struts-core-1.3.8.jar	CVE-2013-2134
struts-core-1.3.8.jar	CVE-2013-2135
struts-core-1.3.8.jar	CVE-2014-0094
struts-core-1.3.8.jar	CVE-2014-0113
struts-core-1.3.8.jar	CVE-2014-0114
struts-core-1.3.8.jar	CVE-2015-0899
struts-core-1.3.8.jar	CVE-2015-5169
struts-core-1.3.8.jar	CVE-2016-0785
struts-core-1.3.8.jar	CVE-2016-1181
struts-core-1.3.8.jar	CVE-2016-1182
struts-core-1.3.8.jar	CVE-2016-4003
struts-taglib-1.3.8.jar	CVE-2012-0394
struts-taglib-1.3.8.jar	CVE-2013-2115
struts-taglib-1.3.8.jar	CVE-2014-0114

struts-taglib-1.3.8.jar	CVE-2015-0899
struts-taglib-1.3.8.jar	CVE-2016-1181
struts-taglib-1.3.8.jar	CVE-2016-1182
struts-tiles-1.3.8.jar	CVE-2012-0394
struts-tiles-1.3.8.jar	CVE-2013-2115
struts-tiles-1.3.8.jar	CVE-2014-0114
struts-tiles-1.3.8.jar	CVE-2015-0899
struts-tiles-1.3.8.jar	CVE-2016-1181
struts-tiles-1.3.8.jar	CVE-2016-1182
jquery.min.js	CVE-2012-6708
jquery.min.js	CVE-2015-9251
jquery.min.js	CVE-2019-11358
jquery-ui.custom.min.js	CVE-2016-7103
jquery-1.3.2.min.js	CVE-2011-4969
jquery-1.3.2.min.js	CVE-2012-6708
jquery-1.3.2.min.js	CVE-2019-11358
jquery-1.3.1.js	CVE-2011-4969
jquery-1.3.1.js	CVE-2012-6708
jquery-1.3.1.js	CVE-2019-11358
jquery-ui-1.7.2.custom.min.js	CVE-2016-7103
handlebars-2.0.0.js	Disallow calling helperMissing and blockHelperMissing directly
handlebars-2.0.0.js	Prototype pollution
handlebars-2.0.0.js	Quoteless attributes in templates can lead to XSS
jquery-1.8.0.min.js	CVE-2012-6708
jquery-1.8.0.min.js	CVE-2015-9251
jquery-1.8.0.min.js	CVE-2019-11358
angular-sanitize.min.js	DOS in $sanitize
angular-sanitize.min.js	Prototype pollution
angular-sanitize.min.js	Universal CSP bypass via add-on in Firefox
angular-sanitize.min.js	XSS in $sanitize in Safari/Firefox
angular-sanitize.min.js	XSS through SVG if enableSvg is set
knockout-2.2.1.js	XSS injection point in attr name binding for browser IE7 and older
jquery-1.12.4.min.js	CVE-2015-9251
jquery-1.12.4.min.js	CVE-2019-11358
moment-with-locales.min.js	reDOS - regular expression denial of service
angular.min.js	DOS in $sanitize
angular.min.js	Prototype pollution
angular.min.js	Universal CSP bypass via add-on in Firefox
angular.min.js	XSS in $sanitize in Safari/Firefox
angular.min.js	XSS through SVG if enableSvg is set
angular-resource.min.js	DOS in $sanitize

angular-resource.min.js	Prototype pollution
angular-resource.min.js	Universal CSP bypass via add-on in Firefox
angular-resource.min.js	XSS in $sanitize in Safari/Firefox
angular-resource.min.js	XSS through SVG if enableSvg is set
moment.min.js	reDOS - regular expression denial of service
angular.js	DOS in $sanitize
angular.js	Prototype pollution
angular.js	Universal CSP bypass via add-on in Firefox
angular.js	XSS in $sanitize in Safari/Firefox
angular.js	XSS through SVG if enableSvg is set
moment.js	reDOS - regular expression denial of service
handlebars.js	Disallow calling helperMissing and blockHelperMissing directly
handlebars.js	Prototype pollution
handlebars.js	Quoteless attributes in templates can lead to XSS
jquery-ui.min.js	CVE-2016-7103
jquery-1.11.1.min.js	CVE-2015-9251
jquery-1.11.1.min.js	CVE-2019-11358
jquery-ui.js	CVE-2016-7103
jquery-1.9.1.min.js	CVE-2015-9251
jquery-1.9.1.min.js	CVE-2019-11358
jquery-1.2.6.min.js	CVE-2011-4969
jquery-1.2.6.min.js	CVE-2012-6708
jquery-1.2.6.min.js	CVE-2019-11358
angular-1.4.5.min.js	DOS in $sanitize
angular-1.4.5.min.js	Prototype pollution
angular-1.4.5.min.js	The attribute usemap can be used as a security exploit
angular-1.4.5.min.js	Universal CSP bypass via add-on in Firefox
angular-1.4.5.min.js	XSS in $sanitize in Safari/Firefox
jquery-2.1.3.min.js	CVE-2015-9251
jquery-2.1.3.min.js	CVE-2019-11358
jquery-ui-1.8.21.custom.min.js	CVE-2010-5312
jquery-ui-1.8.21.custom.min.js	CVE-2016-7103
jquery-1.7.2.min.js	CVE-2012-6708
jquery-1.7.2.min.js	CVE-2015-9251
jquery-1.7.2.min.js	CVE-2019-11358
jquery.js	CVE-2011-4969
jquery.js	CVE-2012-6708
jquery.js	CVE-2019-11358
handlebars.min.js	Disallow calling helperMissing and blockHelperMissing directly
handlebars.min.js	Prototype pollution
handlebars.min.js	Quoteless attributes in templates can lead to XSS

jquery-ui-1.9.2.custom.min.js	CVE-2010-5312
jquery-ui-1.9.2.custom.min.js	CVE-2012-6662
jquery-ui-1.9.2.custom.min.js	CVE-2016-7103
jquery-1.8.3.min.js	CVE-2012-6708
jquery-1.8.3.min.js	CVE-2015-9251
jquery-1.8.3.min.js	CVE-2019-11358
jquery.min.js	CVE-2011-4969
jquery.min.js	CVE-2012-6708
jquery.min.js	CVE-2015-9251
jquery.min.js	CVE-2019-11358
jquery-ui.custom.min.js	CVE-2010-5312
jquery-ui.custom.min.js	CVE-2016-7103
coreapps.vendor.js	CVE-2019-11358
coreapps.vendor.js	DOS in $sanitize
coreapps.vendor.js	Prototype pollution
coreapps.vendor.js	Universal CSP bypass via add-on in Firefox
coreapps.vendor.js	XSS in $sanitize in Safari/Firefox
coreapps.vendor.js	XSS through SVG if enableSvg is set

b) RedHat Victims

(1) Test module

org.springframework:spring-core is vulnerable to CVE-2016-5007
org.apache.poi:poi is vulnerable to CVE-2017-5644
mysql:mysql-connector-java is vulnerable to CVE-2017-3523

(2) Api module

org.apache.poi:poi is vulnerable to CVE-2017-5644
mysql:mysql-connector-java is vulnerable to CVE-2017-3523
commons-beanutils:commons-beanutils is vulnerable to CVE-2014-0114
org.springframework:spring-core is vulnerable to CVE-2016-5007
com.thoughtworks.xstream:xstream is vulnerable to CVE-2013-7285
com.thoughtworks.xstream:xstream is vulnerable to CVE-2017-7957
xerces:xercesImpl is vulnerable to CVE-2009-2625
xerces:xercesImpl is vulnerable to CVE-2013-4002
org.hibernate:hibernate-validator is vulnerable to CVE-2014-3558
com.fasterxml.jackson.core:jackson-databind is vulnerable to CVE-2017-17485
com.fasterxml.jackson.core:jackson-databind is vulnerable to CVE-2017-7525
com.fasterxml.jackson.core:jackson-databind is vulnerable to CVE-2018-5968
com.fasterxml.jackson.core:jackson-databind is vulnerable to CVE-2018-7489

(3) Web module

commons-beanutils:commons-beanutils is vulnerable to CVE-2014-0114
org.springframework:spring-core is vulnerable to CVE-2016-5007
com.thoughtworks.xstream:xstream is vulnerable to CVE-2013-7285
com.thoughtworks.xstream:xstream is vulnerable to CVE-2017-7957
xerces:xercesImpl is vulnerable to CVE-2009-2625
xerces:xercesImpl is vulnerable to CVE-2013-4002
org.hibernate:hibernate-validator is vulnerable to CVE-2014-3558
com.fasterxml.jackson.core:jackson-databind is vulnerable to CVE-2017-17485
com.fasterxml.jackson.core:jackson-databind is vulnerable to CVE-2017-7525
com.fasterxml.jackson.core:jackson-databind is vulnerable to CVE-2018-5968
com.fasterxml.jackson.core:jackson-databind is vulnerable to CVE-2018-7489
org.apache.poi:poi is vulnerable to CVE-2017-5644
commons-fileupload:commons-fileupload is vulnerable to CVE-2013-0248
commons-fileupload:commons-fileupload is vulnerable to CVE-2013-2186
commons-fileupload:commons-fileupload is vulnerable to CVE-2014-0050
commons-fileupload:commons-fileupload is vulnerable to CVE-2016-1000031
commons-fileupload:commons-fileupload is vulnerable to CVE-2016-3092
org.springframework:spring-web is vulnerable to CVE-2015-3192
org.springframework:spring-web is vulnerable to CVE-2015-5211
org.springframework:spring-webmvc is vulnerable to CVE-2015-5211
mysql:mysql-connector-java is vulnerable to CVE-2017-3523

(4) Webapp module

org.apache.poi:poi is vulnerable to CVE-2017-5644
commons-beanutils:commons-beanutils is vulnerable to CVE-2014-0114
org.springframework:spring-core is vulnerable to CVE-2016-5007
com.thoughtworks.xstream:xstream is vulnerable to CVE-2013-7285
com.thoughtworks.xstream:xstream is vulnerable to CVE-2017-7957
xerces:xercesImpl is vulnerable to CVE-2009-2625
xerces:xercesImpl is vulnerable to CVE-2013-4002
org.hibernate:hibernate-validator is vulnerable to CVE-2014-3558
com.fasterxml.jackson.core:jackson-databind is vulnerable to CVE-2017-17485
com.fasterxml.jackson.core:jackson-databind is vulnerable to CVE-2017-7525
com.fasterxml.jackson.core:jackson-databind is vulnerable to CVE-2018-5968
com.fasterxml.jackson.core:jackson-databind is vulnerable to CVE-2018-7489
commons-fileupload:commons-fileupload is vulnerable to CVE-2013-0248
commons-fileupload:commons-fileupload is vulnerable to CVE-2013-2186
commons-fileupload:commons-fileupload is vulnerable to CVE-2014-0050
commons-fileupload:commons-fileupload is vulnerable to CVE-2016-1000031
commons-fileupload:commons-fileupload is vulnerable to CVE-2016-3092
org.springframework:spring-web is vulnerable to CVE-2015-3192
org.springframework:spring-web is vulnerable to CVE-2015-5211

org.springframework:spring-webmvc is vulnerable to CVE-2015-5211
mysql:mysql-connector-java is vulnerable to CVE-2017-3523

(5) Tools module

0 vulnerabilities detected.

c) GitHub's Checker

log4j:log4j - CVE-2019-17571

org.codehaus.jackson:jackson-mapper-asl - CVE-2019-10172

The above dependencies have been identified as vulnerable, but the modules are not specified under which the dependency is reported as vulnerable.

(1) Web module
Not mentioned in tool.

(2) Webapp module
Not mentioned in tool.

(3) Api module
Not mentioned in tool.

(4) Test module
Not mentioned in tool.

(5) Tools module
Not highlighted.

d) Snyk

Multiple vulnerabilities detected for each module but the dependencies with the reported CVE by the tool are as follows:

(1) Web module

Dependency	CVE Reported
com.fasterxml.jackson.core:jackson-databind	CVE-2017-7525

(2) Webapp module

Dependency	CVE Reported
com.fasterxml.jackson.core:jackson-databind	CVE-2017-7525

(3) Api module

Dependency	CVE Reported
com.fasterxml.jackson.core:jackson-databind	CVE-2017-7525

(4) Test module

0 CVEs detected by the tool.

(5) Tools module
0 vulnerabilities detected.

e) Sonatype DepShield
The vulnerable dependencies have been reported as below:
The module for the dependencies is not stated explicitly due to insufficient reporting by the tool.

com.mchange:c3p0:0.9.2.1 : [CVE-2019-5427] Resource Management Errors

org.apache.struts:struts-core:1.3.8 : [CVE-2012-0838] Improper Input Validation , [CVE-2013-2134] Improper Control of Generation of Code ("Code Injection") , [CVE-2012-0391] Improper Input Validation

dom4j:dom4j:1.6.1 : [CVE-2018-1000632] XML Injection

org.springframework:spring-oxm:4.1.4.RELEASE : [CVE-2015-3192] Improper Restriction of Operations within the Bounds of a Memory Buffer

taglibs:standard:1.1.2 : [CVE-2015-0254] Apache Standard Taglibs before 1.2.3 allows remote attackers to execute arbitrar...

org.openmrs.web:openmrs-web:2.4.0-SNAPSHOT : [CVE-2017-12796] Deserialization of Untrusted Data

org.springframework:spring-web:4.1.4.RELEASE : [CVE-2015-5211] Improper Input Validation

org.springframework:spring-core:4.1.4.RELEASE : [CVE-2018-1272] Permissions, Privileges, and Access Controls

org.codehaus.groovy:groovy-all:2.4.6 : [CVE-2016-6814] Deserialization of Untrusted Data

org.codehaus.jackson:jackson-mapper-asl:1.9.13 : [CVE-2018-14718] Deserialization of Untrusted Data , [CVE-2018-7489] Incomplete Blacklist, Deserialization of Untrusted Data , [CVE-2017-7525] Deserialization of Untrusted Data , [CVE-2017-17485] Improper Control of Generation of Code ("Code Injection")

org.openmrs.api:openmrs-api:2.4.0-SNAPSHOT : [CVE-2017-12796] Deserialization of Untrusted Data

commons-beanutils:commons-beanutils:1.9.3 : [CVE-2019-10086] In Apache Commons Beanutils 1.9.2, a special BeanIntrospector class was added wh...

org.springframework:spring-webmvc:4.1.4.RELEASE : (CVSS 8.6) [CVE-2015-5211] Improper Input Validation

javax.servlet:jstl:1.1.2 : [CVE-2015-0254] Apache Standard Taglibs before 1.2.3 allows remote attackers to execute arbitrar...

- For each dependency identified by the tool as having a vulnerability, determine if it is a direct or transitive dependency. A direct dependency is the dependency that is directly accessed by a project. e.g. through declaration in the pom.xml file. A transitive dependency is the dependency of any direct dependency, e.g. not declared in the pom.xml file but a build tool like maven is still able to determine these dependencies and download them for a successful build.

Ans.

a) OWASP Dependency Check
(1) Api module

Dependency	Dependency Type
mysql-connector-java-5.1.28.jar	Direct
postgresql-9.0-801.jdbc4.jar	Direct
commons-beanutils-1.7.0.jar	Direct
log4j-1.2.15.jar	Direct
spring-core-4.1.4.RELEASE.jar	Direct
spring-beans-4.1.4.RELEASE.jar	Direct
spring-context-4.1.4.RELEASE.jar	Direct
spring-expression-4.1.4.RELEASE.jar	Transitive

spring-aop-4.1.4.RELEASE.jar	Direct
spring-tx-4.1.4.RELEASE.jar	Direct
spring-jdbc-4.1.4.RELEASE.jar	Direct
spring-context-support-4.1.4.RELEASE.jar	Direct
xalan-2.7.0.jar	Transitive
c3p0-0.9.2.1.jar	Transitive
dom4j-1.6.1.jar	Direct
xstream-1.4.3.jar	Direct
modify-column-2.0.2.jar	Direct
identity-insert-1.2.1.jar	Direct
type-converter-1.0.1.jar	Direct
xercesImpl-2.8.0.jar	Direct
hibernate-validator-4.2.0.Final.jar	Direct
jackson-mapper-asl-1.9.13.jar	Direct
jackson-databind-2.9.0.jar	Direct
groovy-all-2.4.6.jar	Transitive

(2) Test module

Dependency	Dependency Type
spring-test-4.1.4.RELEASE.jar	Direct
poi-3.5-beta5.jar	Transitive
log4j-1.2.15.jar	Direct
mysql-connector-java-5.1.28.jar	Direct
derbyclient-10.4.2.0.jar	Transitive
postgresql-9.0-801.jdbc4.jar	Transitive

(3) Web module

Dependency	Type
openmrs-api-2.1.3.jar	Direct
commons-beanutils-1.7.0.jar	Direct
log4j-1.2.15.jar	Direct
spring-core-4.1.4.RELEASE.jar	Direct
spring-beans-4.1.4.RELEASE.jar	Direct
spring-context-4.1.4.RELEASE.jar	Direct
spring-aop-4.1.4.RELEASE.jar	Direct
spring-tx-4.1.4.RELEASE.jar	Direct
spring-jdbc-4.1.4.RELEASE.jar	Direct
spring-context-support-4.1.4.RELEASE.jar	Direct

xalan-2.7.0.jar	Transitive
c3p0-0.9.2.1.jar	Transitive
dom4j-1.6.1.jar	Direct
xstream-1.4.3.jar	Direct
modify-column-2.0.2.jar	Direct
identity-insert-1.2.1.jar	Direct
type-converter-1.0.1.jar	Direct
xercesImpl-2.8.0.jar	Direct
hibernate-validator-4.2.0.Final.jar	Direct
jackson-mapper-asl-1.9.13.jar	Direct
jackson-databind-2.9.0.jar	Direct
groovy-all-2.4.6.jar	Transitive
jstl-1.1.2.jar	Direct
commons-fileupload-1.2.1.jar	Direct
spring-web-4.1.4.RELEASE.jar	Direct
spring-webmvc-4.1.4.RELEASE.jar	Direct

(4) Webapp module

Dependency	Type
openmrs-api-2.1.3.jar	Direct
commons-beanutils-1.7.0.jar	Direct
log4j-1.2.15.jar	Direct
spring-core-4.1.4.RELEASE.jar	Direct
spring-beans-4.1.4.RELEASE.jar	Direct
spring-context-4.1.4.RELEASE.jar	Direct
spring-expression-4.1.4.RELEASE.jar	Indirect
spring-aop-4.1.4.RELEASE.jar	Direct
spring-orm-4.1.4.RELEASE.jar	Direct
spring-tx-4.1.4.RELEASE.jar	Direct
spring-jdbc-4.1.4.RELEASE.jar	Direct
spring-context-support-4.1.4.RELEASE.jar	Direct
xalan-2.7.0.jar	Indirect
c3p0-0.9.2.1.jar	Indirect
dom4j-1.6.1.jar	Direct
xstream-1.4.3.jar	Direct
modify-column-2.0.2.jar	Direct
identity-insert-1.2.1.jar	Direct
type-converter-1.0.1.jar	Direct
xercesImpl-2.8.0.jar	Direct
hibernate-validator-4.2.0.Final.jar	Direct
jackson-mapper-asl-1.9.13.jar	Direct

jackson-databind-2.9.0.jar	Direct
groovy-all-2.4.6.jar	Indirect
openmrs-web-2.1.3.jar	Direct
jstl-1.1.2.jar	Direct
commons-fileupload-1.2.1.jar	Direct
spring-web-4.1.4.RELEASE.jar	Direct
spring-webmvc-4.1.4.RELEASE.jar	Direct
spring-oxm-4.1.4.RELEASE.jar	Direct
mysql-connector-java-5.1.28.jar	Direct
postgresql-9.0-801.jdbc4.jar	Direct
standard-1.1.2.jar	Direct
struts-core-1.3.8.jar	Indirect
struts-taglib-1.3.8.jar	Indirect
struts-tiles-1.3.8.jar	Indirect

(5) Tools module
0 vulnerabilities detected.

b) RedHat Victims

org.springframework:spring-core - Direct
org.apache.poi:poi - Indirect
mysql:mysql-connector-java - Direct
commons-beanutils:commons-beanutils - Direct
com.thoughtworks.xstream:xstream - Direct
xerces:xercesImpl - Direct
org.hibernate:hibernate-validator - Direct
com.fasterxml.jackson.core:jackson-databind - Direct
commons-beanutils:commons-beanutils - Direct
commons-fileupload:commons-fileupload - Direct
org.springframework:spring-web - Direct
org.springframework:spring-webmvc - Direct

c) GitHub's Checker
jackson-mapper-asl – Direct
log4j – Direct

d) Snyk

com.fasterxml.jackson.core:jackson-databind – Direct
com.mchange:c3p0 – Indirect
commons-beanutils:commons-beanutils – Direct
dom4j:dom4j – Direct
javax.servlet:jstl – Direct
mysql:mysql-connector-java – Direct
org.apache.struts:struts-core – Indirect
org.codehaus.groovy:groovy-all – Indirect
org.springframework:spring-web – Direct
org.springframework:spring-webmvc – Direct
taglibs:standard – Direct
xerces:xercesImpl – Direct
org.codehaus.jackson:jackson-mapper-asl – Direct
org.springframework:spring-oxm – Direct
org.springframework:spring-core - Direct
com.h2database:h2 – Direct
org.apache.poi:poi-ooxml – Indirect

e) Sonatype Depshield

javax.mail:mail:1.4.1	Direct
com.mchange:c3p0:0.9.2.1	Direct
org.apache.struts:struts-core:1.3.8	Direct
dom4j:dom4j:1.6.1	Direct
org.springframework:spring-oxm:4.1.4.RELEASE	Direct
taglibs:standard:1.1.2	Direct
org.openmrs.web:openmrs-web:2.4.0-SNAPSHOT	Direct
org.springframework:spring-web:4.1.4.RELEASE	Direct
org.springframework:spring-core:4.1.4.RELEASE	Direct
org.codehaus.groovy:groovy-all:2.4.6	Direct
org.codehaus.jackson:jackson-mapper-asl:1.9.13	Direct
org.openmrs.api:openmrs-api:2.4.0-SNAPSHOT	Direct
javax.servlet:jstl:1.1.2	Direct
org.springframework:spring-webmvc:4.1.4.RELEASE	Direct
commons-beanutils:commons-beanutils:1.9.3	Direct

- For up to 10 vulnerable dependencies per tool (report all for tools that report 10 or less, randomly choose 10 for tools that report more than 10), list which vulnerable dependencies have a safer version available.

Ans.

a) OWASP Dependency Check

Dependency	Safer Version
mysql-connector-java-5.1.28.jar	mysql-connector-java-5.1.48.jar
derbyclient-10.4.2.0.jar	derbyclient-10.15.1.3.jar
xalan-2.7.0.jar	xalan-2.7.2.jar
dom4j-1.6.1.jar	dom4j-2.1.1.jar
spring-web-4.1.4.RELEASE.jar	spring-web-5.2.3.RELEASE.jar
jackson-databind-2.9.0.jar	jackson-databind-2.10.2.jar
jackson-mapper-asl-1.9.13.jar	jackson-mapper-asl-2.10.2.jar
xercesImpl-2.8.0.jar	xercesImpl-2.9.1.jar
xstream-1.4.3.jar	xstream-1.4.11.jar
hibernate-validator-4.2.0.Final.jar	hibernate-validator-6.1.2.Final.jar

b) RedHat

Dependency	Safer Version
mysql-connector-java-5.1.28.jar	mysql-connector-java-8.0.19.jar
commons-beanutils-1.7.0.jar	commons-beanutils-1.9.3.jar
xstream-1.4.3.jar	xstream-1.4.11.1.jar
jackson-databind-2.9.0.jar	jackson-databind-2.10.1.jar
commons-fileupload-1.2.1.jar	commons-fileupload-1.4.jar
xercesImpl-2.8.0.jar	xercesImpl-2.11.0.jar
hibernate-validator-4.2.0.Final.jar	hibernate-validator-6.1.0.Final.jar

c) GitHub's Checker

Dependency	Safer Version
log4j-1.2.15.jar	log4j-2.12.1.jar

d) Snyk

Dependency	Safer Version

com.fasterxml.jackson.core:jackson-databind	2.9.10.3
com.mchange:c3p0	0.9.5.3
commons-beanutils:commons-beanutils	1.9.4
mysql:mysql-connector-java	8.0.13
org.codehaus.groovy:groovy-all	2.4.7
org.springframework:spring-web	4.2.2.RELEASE
org.springframework:spring-webmvc	4.3.1.RELEASE
taglibs:standard	1.2.3
xerces:xercesImpl	2.12.0
org.springframework:spring-core	4.1.7.RELEASE

e) Sonatype Depshield

Dependency	Safer Version
org.springframework:spring-oxm:4.1.4.RELEASE	5.2.3.RELEASE
org.openmrs.web:openmrs-web:2.4.0-SNAPSHOT	2.9.10.3
org.springframework:spring-web:4.1.4.RELEASE	5.2.3.RELEASE
org.codehaus.groovy:groovy-all:2.4.6	2.4.7
org.codehaus.jackson:jackson-mapper-asl:1.9.13	2.10.2
javax.servlet:jstl:1.1.2	4.0.1
commons-beanutils:commons-beanutils:1.9.3	1.9.4
org.springframework:spring-webmvc:4.1.4.RELEASE	5.2.3.RELEASE

2. Explain why you think the results differ among the six tools and write a comparison report.

(Hint: your report may explain why a certain tool missed a vulnerable dependency that another tool has detected. What do you think are the strength and weaknesses of each tool from both technical and usability standpoints.)

Ans.

OWASP Dependency Check provides a descriptive report for each module.

- Lists all CVE's for each vulnerable dependency and provides a mapping from NVD.
- The dependencies have vulnerable version numbers as well.
- Categorizes severity of each vulnerable dependency.
- Provides details regarding the vulnerability.
- The report is very descriptive, and the csv file gives in depth detail.
- Fails to identify the safer versions available for the dependency.

RedHat Victims

- Lists vulnerable dependencies followed by the CVE's its categorized under.
- The identified dependencies posing a vulnerability do not provide details of the version number which are affected.
- The report is a simple text file and not detailed.
- Fails to identify the safer versions available for the dependencies.

GitHub's Checker

- Identifies the vulnerable dependencies and lists the CVE's too. Provides a brief description of the vulnerability.
- Highlights the patched versions of the dependencies if available. Provides the vulnerable versions of the dependencies as well.
- Severity of the vulnerable dependencies is highlighted.
- Compared to the other tools the dependencies which are identified as vulnerable are very less.
- The module under which the vulnerability is identified is not highlighted for this tool.

Snyk

- Vulnerable dependencies are identified, and details are also provided as to how they are introduced using different dependencies. Description of the issue is also provided in detail.
- The versions are also provided under which the highlighted exploits are fixed.
- The severity of the issue is also highlighted along with the possible details, the dependency is susceptible to (eg. Arbitrary Code Injection, Access Control Bypass, etc.).
- Compared to other tools, the reports generated using Snyk are very accurate with in depth details regarding each vulnerability.

Sonatype Depshield

- Results report the vulnerabilities and the CVE number associated with it.
- It also highlights whether the vulnerability is direct or transitive.
- The number of identified issues are less compared to other tools.
- The tools fails to identify the module under which the dependency is identified as vulnerable.

1. Log4j

Jar File: log4j-1.2.15.jar

Is Exploitable: Strongly Disagree

Dependency is used in: NA

CVE: CVE-2019-17571

Description - Included in Log4j 1.2 is a SocketServer class that is vulnerable to deserialization of untrusted data which can be exploited to remotely execute arbitrary code when combined with a deserialization gadget when listening to untrusted network traffic for log data. This affects Log4j versions up to 1.2 up to 1.2.17.

Explanation:

As stated in the CVE, the SocketServer class is vulnerable to deserialization of untrusted data which can be exploited to execute arbitrary code remotely. But, this dependency is not being used here to carry out the function as described in the CVE. A SocketServer class is used to listen to a server connection over a port and pass logs to the socket stream. The log4j being used in this context, is used to write logs to the server log file. This is in contrast to the CVE's description. Furthermore, we were able to go through the codebase to verify if the SocketServer class was being used, which is not the case and hence, this strengthens our claim that the vulnerability is not exploitable.

2. mysql-connector-java-5.1.28.jar

Jar: mysql-connector-java-5.1.28.jar

Is Exploitable: Strongly Agree

Dependency is used in: MigrateDataSet.java

api/src/test/java/org/openmrs/test/MigrateDataSet.java

```
47          private static String[] credentials = BaseContextSensitiveTest
48                  .askForUsernameAndPassword("Enter your MYSQL DATABASE username and
    password");
49
50          private static String tempDatabaseName = "junitmigration";
…
97                          System.out.println("Migrating " +
    fileOrDirectory.getAbsolutePath());
98
99                          System.out.println(execMysqlCmd("DROP DATABASE IF EXISTS " +
    tempDatabaseName, null, false));
```

CVE: CVE-2019-2692

Description - Vulnerability in the MySQL Connectors component of Oracle MySQL (subcomponent: Connector/J). Supported versions that are affected are 8.0.15 and prior. Difficult

to exploit vulnerability allows high privileged attacker with logon to the infrastructure where MySQL Connectors executes to compromise MySQL Connectors. Successful attacks require human interaction from a person other than the attacker. Successful attacks of this vulnerability can result in takeover of MySQL Connectors.

Explanation:

As per the issue stated in CVE-2019-2692, the mysql dependency can be used to takeover MySQL connectors which involves human interaction for taking in user input. The line 47 is seen asking the user to input database credentials. Further, these credentials are used to call functions like doMigration() a mysql command is executed directly from the server shell using these credentials. So a user privileges can inject malicious code.

3. Dom4j

jar: dom4j-1.6.1.jar

Is Exploitable: Disagree

Dependency is used in: UpdateFileParser.java, ModuleFilterDefinition.java

CVE: CVE-2018-1000632

Description: dom4j version prior to version 2.1.1 contains a CWE-91: XML Injection vulnerability in Class: Element. Methods: addElement, addAttribute that can result in an attacker tampering with XML documents through XML injection. This attack appear to be exploitable via an attacker specifying attributes or elements in the XML document. This vulnerability appears to have been fixed in 2.1.1 or later.

Explanation:

The dom4j dependency is used to parse a HTML or XML DOM tree into Objects and access their children and their content attributes in Java code. The reported CVE states that the addElement and addAttribute methods of the Element class are exploitable.

This vulnerability poses a threat when there is a create/update operation on the DOM. But, if we see the usage of this dependency in the tool, OpenMRS, it has only been used as a parser, which only accesses existing DOM in Java code and doesn't add any new elements or attributes. A few examples used can be found in the previously mentioned files(UpdateFileParser, SqlDiffFileParser etc.). Hence, this looks like it doesn't impact the application.

4. jackson-mapper-asl-1.9.13.jar

Jar: jackson-mapper-asl-1.9.13.jar

Is Exploitable: Agree Strongly

Dependency is used in:
web/src/main/java/org/springframework/http/converter/json/MappingJacksonHttpMessageConverter.java

CVE: CVE-2017-15095

Description:

A deserialization flaw was discovered in the jackson-databind in versions before 2.8.10 and 2.9.1, which could allow an unauthenticated user to perform code execution by sending the maliciously crafted input to the readValue method of the ObjectMapper. This issue extends the previous flaw CVE-2017-7525 by blacklisting more classes that could be used maliciously.

Explanation:

The jar's function in the project can be viewed as a data binding package which transforms data from one format to other in Java. It converts incoming HTML data into JSON objects. The issue detected says that the readValue method does not sanitize the user input and an unauthenticated user can insert malicious input. The class, MappingJacksonHttpMessageConverter, is seen returning this.objectMapper.readValue(inputMessage.getBody(), javaType); Here, the readValue method of the class is called directly when used with userInputMessage as a method param, without any authentication. As seen, it is a known vulnerability, so it can be treated as a flaw and can be exploited by a smart attacker.

5. Xalan-2.7.0.jar

Jar: xalan-2.7.0.jar

Is Exploitable:

Dependency is used in: WebModuleUtil.java

```
        // write the content into xml file
        TransformerFactory transformerFactory = TransformerFactory.newInstance();
        Transformer transformer = transformerFactory.newTransformer();
        DOMSource source = new DOMSource(doc);
        StreamResult result = new StreamResult(new File(realPath
                + "/WEB-INF/dwr-modules.xml".replace("/", File.separator)));

        transformer.transform(source, result);

    }
    catch (ParserConfigurationException pce) {
        log.error("Failed to parse document", pce);
    }
    catch (TransformerException tfe) {
        log.error("Failed to transorm xml source", tfe);
    }
```

CVE: CVE-2014-0107

Description:

The TransformerFactory in Apache Xalan-Java before 2.7.2 does not properly restrict access to certain properties when FEATURE_SECURE_PROCESSING is enabled, which allows remote attackers to bypass expected restrictions and load arbitrary classes or access external resources via a crafted (1) xalan:content-header, (2) xalan:entities, (3) xslt:content-header, or (4) xslt:entities property, or a Java property that is bound to the XSLT 1.0 system-property function.

Explanation:

WebModuleUtil, OpenmrsUtil classes use TransformerFactory to parse DOM input. But, as per the issue, the vulnerability exists only when the “feature secure processing” flag is set to true. This is disabled by default and the OpenMRS application doesn’t enable it, hence doesn’t pose as a vulnerability.